
SOLO NUESTROS
"QL"
SE EXPLICAN EN ESPAÑOL
DOCUMENTACION EN ESPAÑOL, CONTENIENDO:
• INTRODUCCION • GUIA DEL PRINCIPIANTE • MANUAL DEL SUPERBASIC
• MANUAL TECNICO DE REFERENCIA
• APLICACION "QL" QUILL • APLICACION "QL" ABACUS • APLICACION "QL" ARCHIVE
• APLICACION "QL" EASEL
MICRO WORLD
Modesto Lafuente, 63 Telf. 253 94 54 28003 MADRID
Fuencarral, 100 Telf. 221 23 62 28004 MADRID
Avda. Gaudí, 15 Telf. 256 19 14 08015 BARCELONA
Colombia, 39-41 Telf. 458 61 71 28016 MADRID
Padre Damián, 18 Telf. 259 86 13 28036 MADRID
Stuart, 7 Telf. 891 70 36 ARANJUEZ (Madrid)
José Ortega y Gasset, 21 Telf. 411 28 50 28006 MADRID
Ezequiel González, 28 Telf. 43 68 65 40002 SEGOVIA
MICROHOBBY
REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR
SEMANAL
AÑO II- N.º 14
95 PTAS.
Canarias 105 ptas.
HOP EDITA HOBBY PRESS, S.A.
SOFTWARE
EL SISTEMA TURBO: UNA PROTECCION VULNERABLE
BASIC
TODO SOBRE LOS "BUCLES"
15.000 PTAS. POR TU PROGRAMA
PROGRAMAS
GEOMETRA
TEST
BARQUITOS
INVASORES
LOS "CAZAFANTASMAS": UNA AVENTURA CINEMATOGRAFICA EN TU SPECTRUM

**Director Editorial**
José I. Gómez-Centurión
**Director Ejecutivo**
Domingo Gómez
**Redactor Jefe**
Africa Pérez Tolosa
**Diseño**
Jesús Iniesta
**Maqueta**
Rosa María Capitel
**Redacción**
José María Díaz
Gabriel Nieto
**Colaboradores**
Jesús Alonso, Lorenzo Cebeira, Primitivo de Francisco, Rafael Prades
**Fotografía**
Javier Martínez
Carlos Candel
**Portada**
José María Ponce
**Dibujos**
Manuel Berrocal, J.R. Ballesteros, A. Perera, F.L. Frontán, J. Septien, J.M. López Moreno
**Edita**
HOBBY PRESS, S.A.
**Presidente**
María Andrino
**Consejero Delegado**
José I. Gómez-Centurión
**Administrador General**
Ernesto Marco
**Jefe de Publicidad**
Marisa Esteban
**Secretaría de Publicidad**
Concha Gutiérrez
**Publicidad Barcelona**
Isidro Iglesias
Tel.: (93) 307 11 13
**Secretaría de Dirección**
Marisa Cogorro
**Suscripciones**
M.ª Rosa González
M.ª del Mar Calzada
**Redacción, Administración y Publicidad**
La Granja, n.º 8
Polígono Industrial de Alcobendas
Telf.: 654 32 11
**Dto. Circulación**
Carlos Peropadre
**Distribución**
Coedis, S.A. Valencia, 245. Barcelona.
**Imprime**
Rotedic, S.A.
Carretera de Irún, Km. 12,450
Tel.: 734 15 00
**Fotocomposición**
Consulgraf
Nicolás Morales, 34 - 1.º
Tel.: 471 29 08
**Fotomecánica**
Zescán
Nicolás Morales, 38
Tel.: 472 38 58
**Depósito Legal:**
M-36.598-1984
Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América, 1.532. Telf.: 21 24 64. 1209 BUENOS AIRES (Argentina).





Se solicitará control OJD


# MICROHOBBY ESTA SEMANA


Año II · N.º 14 · 5 al 11 de febrero de 1985
95 ptas. (Sobretasa Canarias 10 ptas.)




# SI NO QUIERE TECLEAR SUS PROGRAMAS, MICROHOBBY LOS GRABA POR USTED:

**CADA MES PONDREMOS A SU DISPOSICION UNA CINTA CON TODOS LOS PROGRAMAS PUBLICADOS EN LOS CUATRO NUMEROS DE DICHO MES.**

La primera cinta contendrá los programas publicados en los números del 1 al 4 inclusive; la segunda, los publicados en los números del 5 al 8, y así sucesivamente.
El precio especial de esta cinta es de 550 ptas., más 75 pesetas por gastos de envío por correo certificado a su domicilio.

SI VD. ESTA INTERESADO EN RECIBIRLA, ESCRIBA A HOBBY PRESS, S.A., APARTADO 54062 DE MADRID, INDICANDO CLARAMENTE QUE MES COMPLETO DE PROGRAMAS DESEA RECIBIR EN CINTA E INCLUYENDO EN EL SOBRE UN TALON NOMINAL A NOMBRE DE HOBBY PRESS, S.A., POR VALOR DE 625 PTAS., O SI LO PREFIERE, EL RESGUARDO DEL GIRO POSTAL A TRAVES DEL CUAL HA EFECTUADO SU PAGO.

**¡ELIJA LA FORMULA QUE MAS LE CONVENGA!**

Cualquier consulta puede realizarla llamando al teléfono: 654 32 11

# MICROPANORAMA




## EL C5 DE SINCLAIR

## UN REVOLUCIONARIO MONOPLAZA ELECTRICO

Sir Clive Sinclair acaba de revolucionar el mercado del transporte con la presentación del C5, un monoplaza electrónico de acabado diseño y fácil uso que es capaz de recorrer 1.600 kilómetros por el módico precio de cinco litros de gasolina.
El modelo en cuestión, estará inmediatamente disponible en el Reino Unido al precio de 399 libras (incluido el IVA) y podrá ser conducido por cualquier persona, a partir de los catorce años, sin ningún tipo de licencia o permiso de circulación.
Totalmente incontaminante, tiene una autonomía de 33 kilómetros por hora (66 con una segunda bateria opcional) y un fácil manejo. El conductor sólo tiene que pulsar un botón para su puesta en marcha, y oprimir una palanca para parar. Durante la noche se recarga la batería mediante un cargador «inteligente», que se enchufa a la red.
Proyectado por Sinclair y desarrollado y probado por Lotus Cars, el C5 debe su funcionamiento a la nueva legislación del Reino Unido, de 1983 para «ciclos ayudados eléctricamente». Un acumulador de plomo de gran rendimiento y un motor eléctrico, impulsan este vehículo que va provisto, también, de unos pedales que aumenta su autonomía y prestaciones, y que permiten su conducción, aún sin batería.
En definitiva, se trata de una gran innovación de pequeño tamaño y gran seguridad que va a convulsionar el mercado y que se distribuirá en establecimientos al por menor en el próximo verano. Su exportación comenzará poco después y entre los mercados de posible exportación se encuentran los de Francia, Italia, Alemania y Holanda.
Ya sabe, por 399 libras, más veintinueve de gastos de envío, podrá conseguir, por ahora tan sólo en el Reino Unido, este maravilloso avance tecnológico «barato de adquisición y mantenimiento, silencioso e incontaminante, confortable y seguro», según reza el Director Gerente de Sinclair Vehicles Limited.

## ZX MICROFERIA

El próximo sábado, día 9 de febrero, tendrá lugar en el Alexandra Palace de Londres, la Feria del ZX que, como todos los años, albergará una buena colección de novedades e innovaciones dentro de este campo de la informática. Como viene siendo habitual en este tipo de exposiciones, a ella acudirán las mejores y más conocidas marcas del mercado internacional que avalarán los más avanzados logros, tanto de hardware como de software, para el Spectrum y el QL.

## UNA MESA PARA TU SPECTRUM



Un nuevo «ingenio» acaba de salir al mercado expresamente diseñado para albergar nuestro ordenador personal y su monitor correspondiente. Se trata de una mesa especial para usuarios de Spectrum que aporta modernidad, utilidad y, sobre todo, comodidad. Está realizado en fibra de vidrio y resinas sintéticas con un sistema de cableado interno que permite, mediante un único interruptor situado bajo el tablero, la conexión y desconexión del ordenador y los periféricos.
Otras características dignas de mención son, por un lado, sus medidas (70 cm. de ancho × 100 de largo × 65 de alto); por otro, el haber sido construida con arreglo a unos cánones de seguridad, de tal manera que el soporte para el monitor es móvil para adaptarse al gusto del usuario.
Se puede decir, pues, que los amantes de la informática están de enhorabuena, porque a partir de ahora las largas horas de dedicación a su hobby serán, cuando menos, más cómodas.

## EL CLUB DEL ORDENADOR PERSONAL

Con este nombre sala a antena semanalmente en Radio Valladolid-Cadena SER, un programa pionero de la radiodifusión española de divulgación que intenta ofrecer unos conocimientos básicos y fundamentales sobre la informática y el mundo de los ordenadores.
El contenido del programa es eminentemente didáctico y experimenta una fórmula totalmente nueva en radio: un curso de «Basic» para principiantes y lecciones sobre el funcionamiento de los ordenadores personales.

## AJEDREZ TRIDIMENSIONAL

La casa Psion, acaba de sacar al mercado un nuevo concepto de ajedrez que incorpora la tercera dimensión y la perspectiva en su visualización dando, de este modo, una mayor autenticidad al juego. Dicho programa, creado específicamente para el QL, ofrece otra serie de ventajas como la de tratarse de uno de los juegos más rápidos, pudiendo efectuar, por ejemplo, más de siete movimientos en un segundo.

Su formato es muy similar a los usados en todos los programas profesionales del QL y, en este caso, los comandos se muestran en la parte superior de la pantalla.

Otro dato a destacar es el de los diferentes niveles de dificultad por los que atraviesa este interesante juego, doce, que pueden ser intercambiados en cualquier momento de la partida. El nivel «O», más indicado para novatos, crea un juego simple pero firme y mueve en frecuencias de segundo, con lo que la rapidez de respuesta es muy elevada.

Para terminar, dos datos más: la posibilidad que ofrece de suspender el juego en cualquier momento de la partida y la movilidad de las piezas por medio de cursores.

## LIBROS




### ZX SPECTRUM QUE ES, PARA QUE SIRVE Y COMO SE USA

Editorial Noray. Dr. Tim Langdell. 212 páginas.

Muchos de nuestros lectores nos preguntan si hay algún libro en el mercado que les sirva para aprender a programar con su Spectrum, e incluso más de uno que no tiene el manual de su ordenador, piden algún sustitutivo de éste. El manual de Tim Langdell, es el más apropiado para ello.
El libro explica, paso a paso, los principios básicos del funcionamiento del Spectrum, desde lo más fundamental, como pueda ser cómo conectar su aparato, hasta una pequeña rutina en código máquina para cambiar los colores de la pantalla.
Según está estructurado, puede servir tanto a los no iniciados, como a aquellos que ya se hayan introducido en el mundo del Spectrum. Es por este motivo, por el que está dividido en dos partes, de las cuales la segunda está pensada para los ya iniciados en el tema.
La primera parte nos va describiendo las principales instrucciones del Basic Sinclair, así como las formas de salvar, cargar y verificar programas. El capítulo del color está muy bien desarrollado y los de gráficos y sonidos están acompañados de numerosos ejemplos, que hacen bastante comprensible los distintos temas. En este sentido, es importante señalar que todo el libro viene con numerosos ejemplos ilustrativos de cada una de las cosas que se van explicando.
La segunda parte es mucho más completa. Además de ampliar los temas de la primera y tratarlos con mucha más profundidad, nos enseña a hacer nuestros propios programas, explicándonos, paso a paso, cómo conseguir los diferentes efectos que más tarde podremos incluir en nuestros juegos. También se tratan los programas de utilidades, e incluso se describen algunos que nos pueden ser muy interesante, como es el caso de un pequeño fichero.
En una última parte del libro se nos describen algunas maneras de mejorar la programación, trucos diversos, una pequeña introducción al código máquina y un repaso general a otros tipos de Basic.




## CAMBIANDO DE CANAL

PRINT # n; «pepe»
Este PRINT, con valores de Ø ó 1, nos permite escribir mensajes en las 2 últimas líneas, los lugares del INPUT. Cuando n = 2 nos escribe la palabra que está entre comillas arriba del todo, y si n = 3 no hace nada.

También podemos incluir la instrucción TAB; basta con poner:

PRINT # 1; TAB 1Ø;«pepe»
y la palabra pepe estará desplazada 1Ø espacios.

Cuando escribamos algo en este lugar hemos de tener en cuenta que es donde aparecen los mensajes de la máquina, y nos puede borrar lo que hayamos escrito al aparecer el mensaje OK.

*Truco enviado por el lector Carlos Pedra Durán*

A. PEREDA

## RAICES DE CUALQUIER ORDEN

En el Spectrum, existe una función para obtener la raíz cuadrada de un número o de una expresión, la funcion SQR.

Sin embargo, no hay una forma directa de obtener las raíces tercera, cuarta, etc...

Una manera de hacerlo es elevar el número cuya raíz queremos obtener, a la potencia inversa del índice, es decir, supongamos que queremos averiguar la raíz cúbica (indice = 3) de 1ØØØØ: escribiremos, por ejemplo PRINT 1ØØØØ↑(1/3) y en general, si el número es m y el índice n, PRINT M↑(1/N).

*Truco enviado por el lector Gabriel García Domingo*

```
  10 INPUT "Indice de la raiz: "
;n
  20 INPUT "Numero: ";m
  30 IF NOT n OR NOT m THEN GO T
O 10
  40 LET r=m↑(1/n): PRINT "Resul
tado: ";r: GO TO 10
```

## CARGANDO CABECERAS

RANDOMIZE USR 1366.
Carga una bandera falsa; si queremos cargar más banderas hemos de poner tantas banderas como instrucciones.

Este RANDOMIZE carga cualquier cabecera, tanto si es la cabecera de un programa en BASIC como si es la cabecera de un programa en código máquina.

*Truco enviado por el lector Carlos Pedra Durán.*

## DETENER UN PROGRAMA

Una de las muchas formas que existen para que el ordenador espere a la pulsación de una tecla y no ejecute ninguna acción hasta ese momento, es la sentencia Basic PAUSE Ø.

Tal vez algunos de nuestros lectores, como nos ha ocurrido a nosotros, se hayan encontrado alguna vez con la desagradable sorpresa de que dicha sentencia no funciona sin causa aparente para ello; el Spectrum simplemente la ignora.

Cuando le suceda esta circunstancia, teclee lo siguiente:

POKE 23611,
(PEEK 23611)-32

y mágicamente se detendrá el programa hasta que se pulse una tecla.

Esta posición de memoria es una de las variables del sistema (FLAGS), que, entre otras cosas, le indica al ordenador si se ha pulsado o no otra tecla.

## PARA ESCRIBIR CON SIGNOS

El Spectrum, para poder escribir los caracteres en pantalla, utiliza un generador de caracteres cuya dirección está contenida en la variable del sistema CHARS (dirección 236Ø6); cambiando los valores de estas dos posiciones de memoria, el ordenador se dirigirá a otra parte de la RAM a la hora de pintar algún caracter alfanumérico.

Si escogemos un valor al azar, normalmente obtendremos una serie de borrones sin ningún sentido; uno de nuestros lectores, Diego López, nos sugiere utilizar POKE 236Ø6, 144 para escribir sólo con signos alfanuméricos, sin «TOKENS».

Para volver a la normalidad, teclear POKE 236Ø6,Ø.

En este espacio también tienen cabida los trucos que nuestros lectores quieran proponer.
Para ello, no tienen más que enviarlos por correo a MICROHOBBY, C/Arzobispo Morcillo, 24, of. 3 y 4, Madrid-28029.

# EDITEXT (y II)

Jesús ALONSO

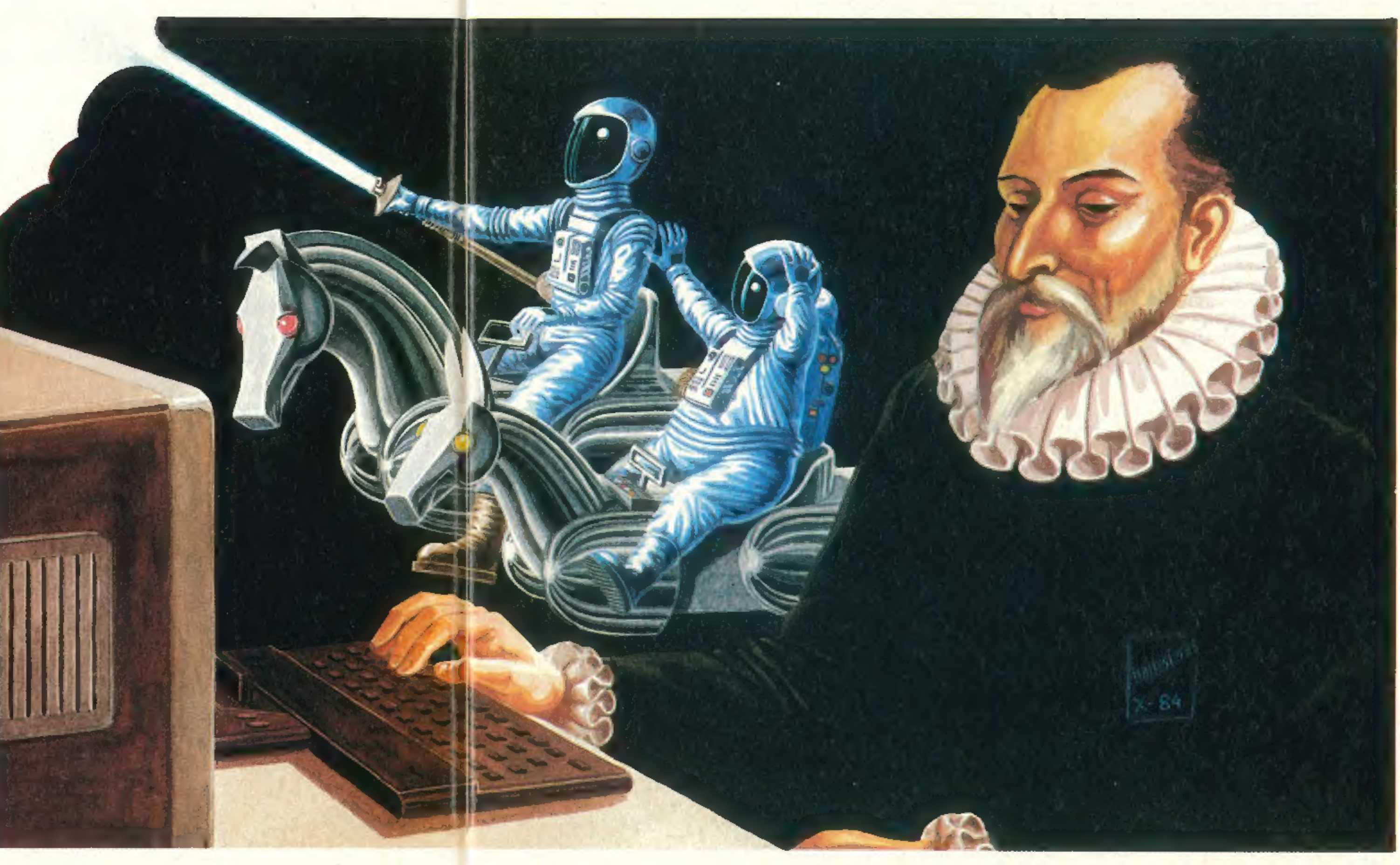
**Spectrum 48 K**

**En el número anterior, ofrecíamos la primera parte de este interesante programa que, por su extensión, hemos querido darlo en dos partes. Pues bien, en esta segunda y última, vamos a ver cómo se carga.**

Para cargar el EDITEXT teclee y ejecute el programa cargador de código máquina; si tod va correctamente aparecerá el mensaje «Start tape, then press any key» y el código máquina, y juego de caracteres se almacenarán en cinta en dos bloques etiquetados con los nombres de fichero «cm» y «cmp»; a continuación haga NEW y teclee el programa en BASIC; cuando lo tenga haga RUN, deberá aparecer la pantalla de «OPCIONES...»; pulse la opción uno, y utilizando el editor, copie la página de instrucciones «COMANDOS DEL EDITEXT», que quedará almacenada en página cero. Cuando termine pulse ENTER en modo extendido, lo que hará salir el mensaje «FIN DE PAGINA», pulse entonces cualquier tecla para ir a «OPCIONES...» y haga BREAK y GO TO 9000, con lo que el programa se guardará en cinta con ejecución automática desde la línea 9000; de esta forma cuando se cargue desde cinta saldrá primero el mensaje «PARE LA CINTA», a continuación la «carátula», y finalmente, la página de instrucciones que se carga en página cero, (por lo que se pierde al empezar a trabajar con el editor). Finalmente, al pulsar cualquier tecla el programa hace RUM (línea 9160) y aparece en primer lugar la pantalla de «OPCIONES...». Por supuesto si detiene el programa durante su ejecución, al volver a hacer RUN no saldrá la carátula, sino que arrancará directamente desde la pantalla de «OPCIONES...».

Si detiene el programa justo durante la impresión de un carácter, es posible que el mensaje «BREAK INTO PROGRAM» le salga con unas letras extrañas, no se preocupe, haga POKE 23606,0: POKE 32607,60 y todo volverá a la normalidad.

**Compruebe con el EDITEXT que en un solo cassette puede escribir un libro.**



## PROGRAMA 2

```
  10 GO TO 8000
 100 POKE 23606,194: POKE 23607,126: LET b=a
 110 PRINT CHR$ a;CHR$ 8; OVER 1;"█";CHR$ 8;: LET cr=cr+1
 120 POKE 23606,0: POKE 23607,60: RETURN
 200 POKE 23606,194: POKE 23607,126: PRINT CHR$ b;CHR$ 8;
 210 POKE 23606,194: POKE 23607,129
 220 PRINT OVER 1;CHR$ a;"█";CHR$ 8;: LET cr=cr+1: GO TO 120
 500 IF cr=0 THEN GO TO 1100
 510 LET m=m-1: LET a=PEEK (m-1): LET cr=cr-2: PRINT " ";CHR$ 8;CHR$ 8;: GO SUB 100: GO TO 1300
 600 LET m=m-1: LET cr=cr-1: PRINT "█";CHR$ 8;: GO TO 1100
 700 LET c=0: PRINT #1;AT 0,28;"    "
 710 IF a=13 THEN PRINT " ": GO TO 1600
 720 IF a=112 OR a=80 THEN LET pa=cr: GO TO 1100
 730 IF a=98 OR a=66 THEN LET pa=0: GO TO 1100
 740 IF a=57 THEN DIM t(4): LET ct=1: GO TO 1100
 742 IF a=49 THEN GO TO 7000
 744 IF a=116 OR a=84 THEN GO TO 7400
 750 LET a=123*(a=97)+124*(a=101)+125*(a=105)+126*(a=111)+127*(a=117): IF a=0 THEN GO TO 1100
 760 GO TO 1170
 800 LET c=0: PRINT #1;AT 0,28;"    "
 810 IF a=13 THEN POKE 23606,194: POKE 23607,126: PRINT CHR$ b: POKE 23606,0: POKE 23607,60: GO TO 1600
 820 IF a=112 OR a=80 THEN LET pa=cr: GO TO 1300
 830 IF a=98 OR a=66 THEN LET pa=0: GO TO 1300
 840 IF a=57 THEN DIM t(4): LET ct=0: GO TO 1300
 842 IF a=49 THEN GO TO 1300
 844 IF a=116 OR a=84 THEN GO TO 7420
 850 LET a=123*(a=97)+124*(a=101)+125*(a=105)+126*(a=111)+127*(a=117): IF a=0 THEN GO TO 1300
 860 GO TO 1370
1000 CLS : INPUT "INTRODUZCA ZONA DE AJUSTE (0-8)";za
1010 IF za<0 OR za>8 THEN GO TO 1000
1020 LET za=64-za: LET c=0: LET pa=0: LET cr=0: LET m=34242: PRINT AT 0,0;"█";CHR$ 8;: BEEP .01,0
1100 PAUSE 0: LET a=CODE INKEY$: BEEP .01,0
1105 IF a=6 THEN GO SUB 1700
1110 IF a<12 THEN GO TO 1100
1120 IF c=1 THEN GO TO 700
1130 IF a=12 THEN GO TO 500
1140 IF a=13 THEN GO TO 1200
1150 IF a=14 THEN LET c=1: PRINT #1;AT 0,28;"EXT.": GO TO 1100
1170 POKE m,a: LET m=m+1: GO SUB 100
1175 IF cr=za THEN BEEP .05,40
1180 IF (a=32 OR a=45) AND cr>=za THEN GO TO 1400
1190 GO TO 1300
1200 IF PEEK 23689=2 THEN GO TO 1600
1210 PRINT " ": IF cr=64 THEN GO TO 1220
1215 POKE m,13: LET m=m+1
1220 LET cr=0
1225 IF pa=cr THEN GO TO 1260
1230 LET a=32: POKE m,a: LET m=m+1: GO SUB 100
1240 IF pa=cr THEN GO TO 1300
1250 LET a=32: POKE m,a: LET m=m+1: GO SUB 200: GO TO 1225
1260 PRINT "█";CHR$ 8;: GO TO 1100
1300 PAUSE 0: LET a=CODE INKEY$: BEEP .01,0
1305 IF a=6 THEN GO SUB 1700
1310 IF a<12 THEN GO TO 1300
1320 IF c=1 THEN GO TO 800
1330 IF a=12 THEN GO TO 600
1340 IF a=13 THEN GO TO 1400
1350 IF a=14 THEN LET c=1: PRINT #1;AT 0,28;"EXT.": GO TO 1300
1370 POKE m,a: LET m=m+1: GO SUB 200
1375 IF cr=za THEN BEEP .05,40
1380 IF (a=32 OR a=45) AND cr>=za THEN GO TO 1200
1390 GO TO 1500
1400 IF PEEK 23689=2 THEN GO TO 1600
1410 POKE 23606,194: POKE 23607,126: PRINT CHR$ b: POKE 23606,0: POKE 23607,60: IF cr=64 THEN GO TO 1220
1415 POKE m,13: LET m=m+1: GO TO 1220
1500 IF PEEK 23689=2 THEN GO TO 1600
1510 GO TO 1100
1600 POKE m,255: BEEP .5,20: PRINT #1;AT 0,0;"FIN DE PAGINA""PULSE PARA IR A OPCIONES": PAUSE 0: CLS : GO TO 8000
1700 POKE 23658,8*(PEEK 23658=0): RETURN
2000 CLS : LET p$="2-ALMACENAR LA PAGINA EN CURSO": LET yy=64: GO SUB 9300
2010 INPUT "NUMERO DE PAGINA? (1-21)";np
2020 POKE 32680,FN a(np): POKE 32681,FN b(np): RANDOMIZE USR 32679
2030 GO TO 8000
3000 CLS : LET p$="3-IMPRIMIR LA PAGINA EN CURSO": LET yy=80: GO SUB 9300
3010 PAUSE 50: CLS
3020 OVER 1: RANDOMIZE USR 32595: OVER 0: POKE 23606,0: POKE 23607,60: COPY
3030 GO TO 8000
4000 CLS : LET p$="4-SALVAR EN CINTA": LET yy=96: GO SUB 9300
4010 INPUT "DESDE QUE PAGINA? (1-21)";np
4020 IF np<1 OR np>21 THEN GO TO 4010
4030 INPUT "CUANTAS PAGINAS? (MAX. 21)";npp
4040 IF (34242+np*1408+npp*1408)>65218 THEN GO TO 4030
4050 PRINT #1;AT 0,0;"Cinta en ""REC"" , pulse una tecla"
4060 SAVE "texto"CODE 34242+np*1408,npp*1408
4070 INPUT "DESEA VERIFICAR? (s/n)"; LINE r$
4080 IF r$(1)="n" OR r$(1)="N" THEN GO TO 8000
4090 VERIFY "texto"CODE 34242+np*1408,npp*1408: GO TO 8000
5000 CLS : LET p$="5-CARGAR DE CINTA": LET yy=112: GO SUB 9300
5010 PRINT AT 18,0;"EL TEXTO EN CINTA SE CARGARA      A PARTIR DE LA PAGINA 1"
5020 PRINT #1;AT 0,0;"Cinta en ""PLAY"", pulse una tecla": PAUSE 0
5025 PRINT #1;AT 0,0;"                                "
5030 LOAD "texto"CODE 35650
5040 GO TO 8000
6000 CLS : LET p$="6-LEER LAS PAGINAS ALMACENADAS": LET yy=128: GO SUB 9300
6010 INPUT "A PARTIR DE QUE PAGINA? (1-21)";np
6020 IF np<1 OR np>21 THEN GO TO 6010
6030 CLS : POKE 32692,FN a(np): POKE 32693,FN b(np): OVER 1: RANDOMIZE USR 32691: OVER 0: POKE 23606,0: POKE 23607,60
6040 PRINT #1;AT 0,0;"PAGINA ";np;AT 1,0;"S:Seguir P:Imprimir: O:Opciones"
6050 PAUSE 0: LET rp=CODE INKEY$
6060 IF rp=79 OR rp=111 THEN GO TO 8000
6070 IF rp=80 OR rp=112 THEN COPY : GO TO 6050
6080 IF rp<>83 AND rp<>115 THEN GO TO 6050
6090 LET np=np+1: IF np>21 THEN GO TO 8000
6100 GO TO 6030
7000 IF cr>0 THEN GO TO 1100
7010 DIM t(4): LET ct=1
7020 PAUSE 0: LET a=CODE INKEY$
7030 IF a=32 THEN GO SUB 100: GO TO 7070
7040 IF a=116 OR a=84 THEN LET t(ct)=cr: LET ct=ct+1: GO TO 7070
7050 IF a=13 THEN GO TO 7200
7060 GO TO 7020
7070 IF ct=5 THEN GO TO 7206
7100 PAUSE 0: LET a=CODE INKEY$
7110 IF a=32 THEN GO SUB 200: GO TO 7150
7120 IF a=116 OR a=84 THEN LET t(ct)=cr: LET ct=ct+1: GO TO 7150
7130 IF a=13 THEN GO TO 7200
7140 GO TO 7100
7150 IF ct=5 THEN GO TO 7200
7160 GO TO 7020
7200 PRINT " ";AT 24-PEEK 23689,0;"█";CHR$ 8;: LET cr=0: GO TO 1100
7400 LET a=32: POKE m,a: LET m=m+1: GO SUB 100
7410 IF cr=t(1) OR cr=t(2) OR cr=t(3) OR cr=t(4) THEN GO TO 1300
7415 IF cr=64 THEN LET cr=0
7420 LET a=32: POKE m,a: LET m=m+1: GO SUB 200
7430 IF cr=t(1) OR cr=t(2) OR cr=t(3) OR cr=t(4) THEN GO TO 1100
7435 IF cr=64 THEN LET cr=0
7440 GO TO 7400
8000 CLS : RANDOMIZE USR 65218
8010 LET xx=16: LET xs=1: LET ys=2
8020 DIM t(4)
8070 DEF FN a(x)=(34242+x*1408)-256*INT ((34242+x*1408)/256)
8080 DEF FN b(x)=INT ((34242+x*1408)/256)
8100 PAUSE 0: LET op=CODE INKEY$: GO TO 8100+10*(op>48 AND op<55)
8110 GO TO (op-48)*1000
9000 CLEAR 32255
9010 PAPER 0: BORDER 1: CLS
9015 PRINT INK 5;AT 10,4;"CARGANDO CODIGO MAQUINA"
9016 INK 0
9020 LOAD "cm"CODE 32256,3394
9025 LOAD "cmp"CODE 65218,292
9030 CLS : PAPER 5: INK 2: FLASH 1
9040 LET p$="PARE LA CINTA": LET xs=2: LET ys=3: LET yy=64: GO SUB 9200
9050 PAUSE 100
9060 PAPER 0: FLASH 0: CLS : PAPER 3: INK 7
9070 LET p$="EDITOR": LET xs=3: LET yy=16: GO SUB 9200
9080 LET p$="  DE  ": LET yy=40: GO SUB 9200
9090 LET p$="TEXTOS": LET yy=64: GO SUB 9200
9100 PAPER 0: INK 6
9110 LET p$="© J. Alonso 1984": LET xs=2: LET yy=104: GO SUB 9200
9120 PAPER 5: INK 2: FLASH 1
9130 LET p$="S-15": LET yy=144: GO SUB 9200
9140 PAUSE 250: BORDER 5: PAPER 1: INK 7: FLASH 0: CLS
9150 OVER 1: RANDOMIZE USR 32595
9155 POKE 23606,0: POKE 23607,60
9160 PAUSE 0: OVER 0: RUN
9200 LET xx=(256-8*xs*LEN p$)/2
9300 LET i=23306: POKE i,xx: POKE i+1,yy: POKE i+2,xs: POKE i+3,ys: POKE i+4,8: LET i=i+4: LET w=LEN p$: FOR n=1 TO w: POKE i+n,CODE p$(n): NEXT n: POKE i+w+1,255: LET w=USR 32256: RETURN
9900 SAVE "editext" LINE 9000
9910 SAVE "cm"CODE 32256,3394
9915 SAVE "cmp"CODE 65218,292
9920 VERIFY "editext"
9930 VERIFY "cm"CODE 32256,3394
9935 VERIFY "cmp"CODE 65218,292
```

## PROGRAMA 3

```
   1 REM MODIFICACIONES AL CARGADOR DE CODIGO MAQUINA DEL PROGRAMA "EDITEXT" PARA TRABAJAR -CON IMPRESORAS QUE UTILICEN LA INTERFACE "INDESCOMP".
Teclee las siguientes lineas en sustitucion de las que llevan el mismo numero en el programa original.(Se relocaliza la ultima rutina maquina 1408 bytes mas abajo).
 510 SAVE "cmp"CODE 63810,292
 540 VERIFY "cmp"CODE 63810,292
4005 DATA 63810,291
4010 DATA "21A5F9110A5B010E00EDB0CD007E21B3F9110A5B011900EDB0CD007E21CCF9110A5B012400"
4020 DATA "EDB0CD007E21F0F9110A5B012300EDB0CD007E2113FA110A5B011700EDB0CD007E212AFA11"
4030 DATA "0A5B011700EDB0CD007E2141FA110A5B012400EDB0CD007EC940080203084F5043494F4E45"
4090 DATA 21991
```

El conocido escritor de ciencia-ficción Arthur C. Clark (autor entre otras, de «2001, una odisea del espacio», escribió su último libro «2010, odisea dos» utilizando un micro-computador «Archives II» con software «Wordstar», y remitió el texto a su editor sobre un «diskete» de 12 centímetros. Espero que el EDITEXT aunque bastante más modesto, permita a los lectores estar a la altura de esta época; si le gusta escribir comprobará que en una cassette le cabe un libro entero.

## MAPA DEL PROGRAMA BASIC

| | |
|---|---|
| 100 | Subrrutina de impresión de caracteres impares. |
| 200 | Subrrutina de impresión de caracteres pares. |
| 300 | Conversión de caracteres dobles (m M w W) impares. |
| 400 | Conversión de caracteres dobles (m M w W) pares. |
| 500 | Borrado de caracteres pares. |
| 600 | Borrado de caracteres impares. |
| 700 | Salto a extendido en carácter impar. |
| 800 | Salto a extendido en carácter par. |
| 1000 | Opción uno, escribir un texto (editor de página en curso). |
| 2000 | Opción dos, almacenar la página en curso. |
| 3000 | Opción tres, imprimir la página en curso. |
| 4000 | Opción cuatro, salvar en cinta. |
| 5000 | Opción cinco, cargar de cinta. |
| 6000 | Opción seis, leer las páginas almacenadas. |
| 7000 | Rutina de tabulado. |
| 7400 | Control de tabulador. |
| 8000 | Rutina de opciones. |
| 9000 | Rutina de presentación. |
| 9200 | Impresión centrada. |
| 9300 | Impresión sin centrar. |
| 9900 | Rutina de carga en cinta. |

## PROGRAMA 4

```
   1 REM  MODIFICACIONES AL PRO-
GRAMA "EDITEXT" PARA TRABAJAR  -
CON IMPRESORAS QUE UTILICEN LA -
INTERFACE "INDESCOMP".
Teclee las siguientes lineas en
sustitucion de las que llevan el
mismo numero en el programa ori-
ginal. (La capacidad de almace-
namiento se reduce a 20 paginas)
  10 RANDOMIZE USR 64973: LPRINT
 CHR$ 14;"   ": GO TO 8000
2010 INPUT "NUMERO DE PAGINA ? (
1-20)";np: IF np<1 OR np>20 THEN
 GO TO 2010
3020 OVER 1: RANDOMIZE USR 32595
: OVER 0: POKE 23606,0: POKE 236
07,60: RANDOMIZE USR 65044
4010 INPUT "DESDE QUE PAGINA ? (
1-20)";np
4020 IF np≤1 OR np>20 THEN GO TO
 4010
4030 INPUT "CUANTAS PAGINAS ? (M
AX. ";(21-np);")";npp
4040 IF (34242+np*1408+npp*1408)
>63810 THEN GO TO 4030
6010 INPUT "A PARTIR DE QUE PAGI
NA ? (1-21)";np
6020 IF np<1 OR np>20 THEN GO TO
 6010
6070 IF rp=80 OR rp=112 THEN INP
UT PI: RANDOMIZE USR 65044: GO T
O 6030
6090 LET np=np+1: IF np>20 THEN
GO TO 8000
8000 CLS : RANDOMIZE USR 63810
```

## RUTINA PARA TRANSFERIR CUALQUIER PAGINA DE ALMACENAMIENTO A LA PAGINA CERO, E IMPRIMIRLA EN PANTALLA

ENTRADAS: En las direcciones 32692 y 32693 se introduce la dirección de la página origen (Cuando se carga, estos valores quedan inicializados para tomar como origen la página uno).
SALIDAS: La página origen se transfiere a la página cero, y se imprime.
LONGITUD: 15 Bytes.
DIRECCION DE ARRANQUE: 32691.
DIRECCION DE INICIO: 32691.
TIEMPO DE EJECUCION: 1 seg.

| Direcc. | Código Máquina | Listado | Assembler | |
|---|---|---|---|---|
| 32691 | 33,66,139 | Ø1Ø | LD HL,8B42 | ;Carga origen |
| 32694 | 17,194,133 | Ø2Ø | LD DE,85C2 | ;Carga destino |
| 32697 | 1,128,5 | Ø3Ø | LD BC,Ø58Ø | ;Carga longitud |
| 327ØØ | 237,176 | Ø4Ø | LDIR | ;Transfiere, incrementa y repite |
| 327Ø2 | 2Ø5,83,127 | Ø5Ø | CALL 7F53 | ;Llama rutina de imp. |
| 327Ø5 | 2Ø1 | Ø6Ø | RET | ;Retorna |

## RUTINA DE ACCESO A LA IMPRESION DE CARACTERES GRANDES PARA EL MENSAJE «OPCIONES...»

ENTRADAS: Lista de datos a partir de 65317.
SALIDAS: Impresión de una pantalla.
LONGITUD: 292 Bytes (incluyendo la lista de datos).
DIRECCION DE ARRANQUE: 65218.
DIRECCION DE INICIO: 65218.
TIEMPO DE EJECUCION: 3 seg.

| Direcc. | Código Máquina | Listado | Assembler | |
|---|---|---|---|---|
| 65218 | 33,37,255 | Ø1Ø | LD HL,FF25 | ;Carga puntero origen |
| 65221 | 17,1Ø,91 | Ø2Ø | LD DE,5BØA | ;Carga puntero destino |
| 65224 | 1,14,Ø | Ø3Ø | LD BC,ØØØE | ;Carga número de Bytes |
| 65227 | 237,176 | Ø4Ø | LDIR | ;Bucle de transferencia |
| 65229 | 2Ø5,Ø,126 | Ø5Ø | CALL 7EØØ | ;Llama rutina de impresión |
| 65232 | 33,51,255 | Ø6Ø | LD HL,FF33 | |
| 65235 | 17,1Ø,91 | Ø7Ø | LD DE,5BØA | |
| 65238 | 1,25,Ø | Ø8Ø | LD BC,ØØ19 | |
| 65241 | 237,176 | Ø9Ø | LDIR | |
| 65243 | 2Ø5,Ø,126 | 1ØØ | CALL 7EØØ | |
| 65246 | 33,76,255 | 11Ø | LD HL,FF4C | |
| 65249 | 17,1Ø,91 | 12Ø | LD DE,5BØA | |
| 65252 | 1,36,Ø | 13Ø | LD BC,ØØ24 | |
| 65255 | 237,176 | 14Ø | LDIR | |
| 65257 | 2Ø5,Ø,126 | 15Ø | CALL 7EØØ | |
| 6526Ø | 33,112,255 | 16Ø | LD HL,FF7Ø | |
| 65263 | 17,1Ø,91 | 17Ø | LD DE,5BØA | |
| 65266 | 1,35,Ø | 18Ø | LD BC,ØØ23 | |
| 65269 | 237,176 | 19Ø | LDIR | |
| 65271 | 2Ø5,Ø,126 | 2ØØ | CALL 7EØØ | |
| 65274 | 33,147,255 | 21Ø | LD HL,FF93 | |
| 65277 | 17,1Ø,91 | 22Ø | LD DE,5BØA | |
| 6528Ø | 1,23,Ø | 23Ø | LD BC,ØØ17 | |
| 65283 | 237,176 | 24Ø | LDIR | |
| 65285 | 2Ø5,Ø,126 | 25Ø | CALL 7EØØ | |
| 65288 | 33,17Ø,255 | 26Ø | LD HL,FFAA | |
| 65291 | 17,1Ø,91 | 27Ø | LD DE,5BØA | |
| 65294 | 1,23,Ø | 28Ø | LD BC,ØØ17 | |
| 65297 | 237,176 | 29Ø | LDIR | |
| 65299 | 2Ø5,Ø,126 | 3ØØ | CALL 7EØØ | |
| 653Ø2 | 33,193,255 | 31Ø | LD HL,FFC1 | |
| 653Ø5 | 17,1Ø,91 | 32Ø | LD DE,5BØA | |
| 653Ø8 | 1,36,Ø | 33Ø | LD BC,ØØ24 | |
| 65311 | 237,176 | 34Ø | LDIR | |
| 65313 | 2Ø5,Ø,126 | 35Ø | CALL 7EØØ | |
| 65316 | 2Ø1 | 36Ø | RET | |

## LISTA DE DATOS PARA LA RUTINA DEL MENSAJE «OPCIONES...»

| Direcc. | Datos | Direcc. | Datos |
|---|---|---|---|
| 65317 | 64,8,2,3 | 65413 | 71,73,78,65 |
| 65321 | 8,79,8Ø,67 | 65417 | 32,69,78,32 |
| 65325 | 73,79,78,69 | 65421 | 67,85,82,83 |
| 65329 | 83,255,16,48 | 65425 | 79,255,16,96 |
| 65333 | 1,2,8,49 | 65429 | 1,2,8,52 |
| 65337 | 45,69,83,67 | 65433 | 45,83,65,76 |
| 65341 | 82,73,66,73 | 65437 | 86,65,82,32 |
| 65345 | 82,32,85,78 | 65441 | 69,78,32,67 |
| 65349 | 32,84,69,88 | 65445 | 73,78,84,65 |
| 65353 | 84,79,255,16 | 65449 | 255,16,112,1 |
| 65357 | 64,1,2,8 | 65453 | 2,8,53,45 |
| 65361 | 5Ø,45,65,76 | 65457 | 67,65,82,71 |
| 65365 | 77,65,67,69 | 65461 | 65,82,32,68 |
| 65369 | 78,65,82,32 | 65465 | 69,32,67,73 |
| 65373 | 76,65,32,8Ø | 65469 | 78,84,65,255 |
| 65377 | 65,71,73,78 | 65473 | 16,128,1,2 |
| 65381 | 65,32,69,78 | 65477 | 8,54,45,76 |
| 65385 | 32,67,85,82 | 65481 | 69,69,82,32 |
| 65389 | 83,79,255,16 | 65485 | 76,65,83,32 |
| 65393 | 8Ø,1,2,8 | 65489 | 8Ø,65,71,73 |
| 65397 | 51,45,73,77 | 65493 | 78,65,83,32 |
| 654Ø1 | 8Ø,82,73,77 | 65497 | 65,76,77,65 |
| 654Ø5 | 73,82,32,76 | 655Ø1 | 67,69,78,65 |
| 654Ø9 | 65,32,8Ø,65 | 655Ø5 | 68,65,83,255 |



# BARQUITOS

J.M. RDGUEZ. CORDOVILLA

**Spectrum 48 K**

**Este juego, estamos seguros, que no necesita grandes explicaciones por su popularidad y antigüedad. Muchos de nosotros hemos pasado largas horas intentando salvar nuestra flota del ataque enemigo.**

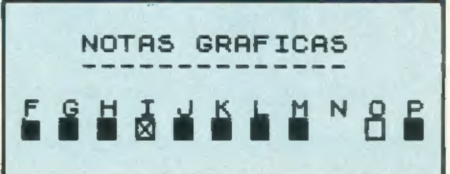



Pues bien, el juego de los barquitos también se ha modernizado y ahora podemos utilizar nuestro ordenador para inventar guerras en alta mar y demostrar nuestra visión estratega.

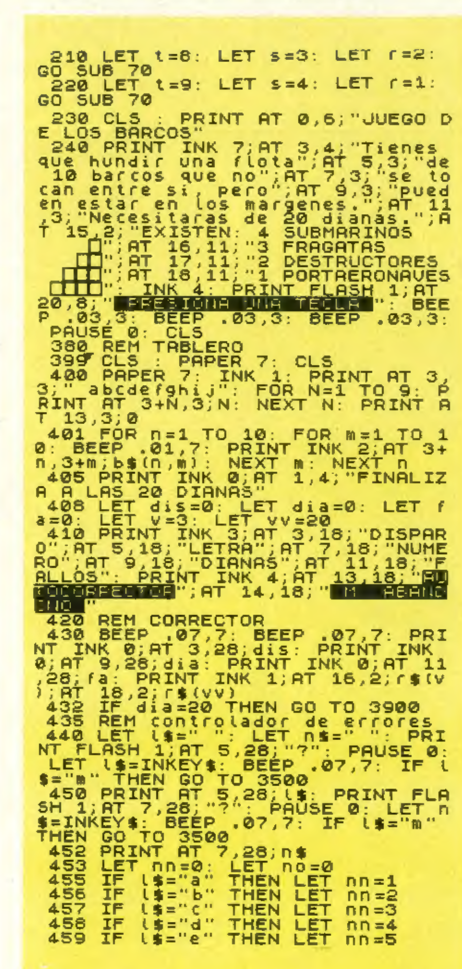
Pero como toda innovación, han introducido algunos pequeños detalles que contribuyen, sin lugar a dudas, a perfeccionar el juego.

Para empezar, el programa puede ser autodestruido a la menor indicación por nuestra parte (tecla 2); está también capacitado para obtener una copia de sí mismo en cinta, y, por último, está confeccionado para comenzar automáticamente justo después de la carga.

Anímese y no se arrepentirá.




```
   1 PAPER 2
   2 CLS : BORDER 2: PAPER 2: IN
K 6: PRINT FLASH 1;AT 8,7;"PARE
EL CASSETTE"
  10 DIM a$(10,10): DIM b$(10,10
): DIM r$(20,25)
  11 PAPER 2
  12 GO SUB 8000
  13 GO TO 4045
  14 BORDER 2: PAPER 2: CLS : IN
K 6: PRINT FLASH 1;AT 11,6;"AGUA
RDE POR FAVOR": PRINT AT 21,12;"
© J.M.RODRIGUEZ 1984"
  15 FOR n=1 TO 10: FOR m=1 TO 1
0: LET a$(n,m)="□": NEXT m: NEXT
 n
  16 GO SUB 8040
  30 RANDOMIZE : LET a4=INT (RND
*2): LET b4=INT (RND*6+1): LET c
4=INT (RND*10+1)
  40 IF a4=1 THEN GO TO 60
  50 LET y4=c4: FOR n=1 TO 4: LE
T x4=b4+n-1: LET a$(x4,y4)="■":
NEXT n: GO TO 200
  60 LET x4=c4: FOR n=1 TO 4: LE
T y4=b4+n-1: LET a$(x4,y4)="■":
NEXT n: GO TO 200
  70 LET b=0
  80 REM COLOCACION DEL RESTO
 100 RANDOMIZE : LET a3=INT (RND
*2): LET b3=INT (RND*t+1): LET c
3=INT (RND*10+1)
 110 IF a3=1 THEN GO TO 130
 120 LET y3=c3: LET yy3=c3-1: LE
T yyy3=c3+1: LET x3=b3: LET xx3=
b3-1: LET xxx3=b3+r+1: GO TO 140
 130 LET x3=c3: LET xx3=c3-1: LE
T xxx3=c3+1: LET y3=b3: LET yy3=
b3-1: LET yyy3=b3+r+1
 140 IF xx3=0 THEN LET xx3=1
 142 IF yy3=0 THEN LET yy3=1
 150 IF xxx3=11 THEN LET xxx3=10
 152 IF yyy3=11 THEN LET yyy3=10
 160 FOR n=xx3 TO xxx3: FOR m=yy
3 TO yyy3
 162 IF a$(n,m)="■" THEN GO TO 1
00
 164 NEXT m: NEXT n
 166 IF a3=1 THEN GO TO 170
 167 LET y3=c3: LET x3=b3
 168 FOR n=1 TO r: LET a$(x3+n,y
3)="■": NEXT n
 169 GO TO 180
 170 LET x3=c3: LET y3=b3
 172 FOR n=1 TO r: LET a$(x3,y3+
n)="■": NEXT n
 180 LET b=b+1: IF b=s THEN RETU
RN
 190 GO TO 100
 199 REM CONTROLADOR
 200 LET t=7: LET s=2: LET r=3:
GO SUB 70
```



```
 210 LET t=8: LET s=3: LET r=2:
GO SUB 70
 220 LET t=9: LET s=4: LET r=1:
GO SUB 70
 230 CLS : PRINT AT 0,6;"JUEGO D
E LOS BARCOS"
 240 PRINT INK 7;AT 3,4;"Tienes
que hundir una flota";AT 5,3;"de
 10 barcos que no";AT 7,3;"se to
can entre si, pero";AT 9,3;"pued
en estar en los margenes.";AT 11
,3;"Necesitaras de 20 dianas.";A
T 15,2;"EXISTEN: 4 SUBMARINOS
    □";AT 16,11;"3 FRAGATAS
   □□";AT 17,11;"2 DESTRUCTORES
  □□□";AT 18,11;"1 PORTAERONAVES
 □□□□"; INK 4: PRINT FLASH 1;AT
20,8;" PRESIONA UNA TECLA ": BEE
P .03,3: BEEP .03,3: BEEP .03,3:
 PAUSE 0: CLS
 380 REM TABLERO
 399 CLS : PAPER 7: CLS
 400 PAPER 7: INK 1: PRINT AT 3,
3;" abcdefghij": FOR N=1 TO 9: P
RINT AT 3+N,3;N: NEXT N: PRINT A
T 13,3;0
 401 FOR n=1 TO 10: FOR m=1 TO 1
0: BEEP .01,7: PRINT INK 2;AT 3+
n,3+m;b$(n,m): NEXT m: NEXT n
 405 PRINT INK 0;AT 1,4;"FINALIZ
A A LAS 20 DIANAS"
 408 LET dis=0: LET dia=0: LET f
a=0: LET v=3: LET vv=20
 410 PRINT INK 3;AT 3,18;"DISPAR
O";AT 5,18;"LETRA";AT 7,18;"NUME
RO";AT 9,18;"DIANAS";AT 11,18;"F
ALLOS": PRINT INK 4;AT 13,18;"AU
TOCORRECTOR";AT 14,18;" M  ABAND
ONO "
 420 REM CORRECTOR
 430 BEEP .07,7: BEEP .07,7: PRI
NT INK 0;AT 3,28;dis: PRINT INK
0;AT 9,28;dia: PRINT INK 0;AT 11
,28;fa: PRINT INK 1;AT 16,2;r$(v
);AT 18,2;r$(vv)
 432 IF dia=20 THEN GO TO 3900
 435 REM controlador de errores
 440 LET l$=" ": LET n$=" ": PRI
NT FLASH 1;AT 5,28;"?": PAUSE 0:
 LET l$=INKEY$: BEEP .07,7: IF l
$="m" THEN GO TO 3500
 450 PRINT AT 5,28;l$: PRINT FLA
SH 1;AT 7,28;"?": PAUSE 0: LET n
$=INKEY$: BEEP .07,7: IF l$="m"
THEN GO TO 3500
 452 PRINT AT 7,28;n$
 453 LET nn=0: LET no=0
 455 IF l$="a" THEN LET nn=1
 456 IF l$="b" THEN LET nn=2
 457 IF l$="c" THEN LET nn=3
 458 IF l$="d" THEN LET nn=4
 459 IF l$="e" THEN LET nn=5
```

```
 460 IF l$="f" THEN LET nn=6
 461 IF l$="g" THEN LET nn=7
 462 IF l$="h" THEN LET nn=8
 463 IF l$="i" THEN LET nn=9
 464 IF l$="j" THEN LET nn=10
 465 IF nn=0 THEN GO TO 420
 466 IF n$="1" THEN LET no=1
 467 IF n$="2" THEN LET no=2
 468 IF n$="3" THEN LET no=3
 469 IF n$="4" THEN LET no=4
 470 IF n$="5" THEN LET no=5
 471 IF n$="6" THEN LET no=6
 472 IF n$="7" THEN LET no=7
 473 IF n$="8" THEN LET no=8
 474 IF n$="9" THEN LET no=9
 475 IF n$="0" THEN LET no=10
 476 IF no=0 THEN GO TO 420
 490 IF b$(no,nn)="■" THEN GO TO
 501
 495 IF b$(no,nn)="⊠" THEN GO TO
 501
 500 GO TO 510
 501 LET i=INT (RND*8+1): IF i=1
 THEN GO SUB 5000
 502 IF i=2 THEN GO SUB 5001
 503 IF i=3 THEN GO SUB 5002
 504 IF i=4 THEN GO SUB 5003
 505 IF i=5 THEN GO SUB 5004
 506 IF i=6 THEN GO SUB 5005
 507 IF i=7 THEN GO SUB 5006
 508 IF i=8 THEN GO SUB 5007
 509 PRINT AT 16,2;r$(v);AT 18,2
;r$(vv): BEEP .7,7: GO TO 420
 510 IF a$(no,nn)="■" THEN GO TO
 600
 515 LET i=INT (RND*27+1): IF i=
1 THEN GO SUB 5010
 516 IF i=2 THEN GO SUB 5011
 517 IF i=3 THEN GO SUB 5012
 518 IF i=4 THEN GO SUB 5013
 519 IF i=5 THEN GO SUB 5014
 520 IF i=6 THEN GO SUB 5015
 521 IF i=7 THEN GO SUB 5016
 522 IF i=8 THEN GO SUB 5017
 523 IF i=9 THEN GO SUB 5018
 524 IF i=10 THEN GO SUB 5019
 525 IF i=11 THEN GO SUB 5020
 526 IF i=12 THEN GO SUB 5021
 527 IF i=13 THEN GO SUB 5022
 528 IF i=14 THEN GO SUB 5023
 529 IF i=15 THEN GO SUB 5024
 530 IF i=16 THEN GO SUB 5025
 531 IF i=17 THEN GO SUB 5026
 532 IF i=18 THEN GO SUB 5027
 533 IF i=19 THEN GO SUB 5028
 534 IF i=20 THEN GO SUB 5029
 535 IF i=21 THEN GO SUB 5030
 536 IF i=22 THEN GO SUB 5031
 537 IF i=23 THEN GO SUB 5032
 538 IF i=24 THEN GO SUB 5033
 539 IF i=25 THEN GO SUB 5034
 540 IF i=26 THEN GO SUB 5035
```

```
 550 LET b$(no,nn)="⊠": LET dis=
dis+1: LET fa=fa+1: PRINT INK 5;
AT 3+no,3+nn;b$(no,nn): PRINT AT
 16,2;r$(v);AT 18,2;r$(vv): BEEP
 .8,8: BEEP .8,7: BEEP .8,6: GO
TO 420
 600 LET dis=dis+1: LET dia=dia+
1: LET xx=no-1: LET yy=nn-1: LET
 xxx=no+1: LET yyy=nn+1
 605 LET b$(no,nn)="█": PRINT IN
K 0;AT 3+no,3+nn;b$(no,nn)
 610 REM correccion normal-subma
rinos
 620 IF xx=0 OR yy=0 THEN GO TO
640
 630 LET b$(xx,yy)="⊠": PRINT IN
K 5;AT 3+xx,3+yy;b$(xx,yy)
 640 IF xx=0 OR yyy=11 THEN GO T
O 660
 650 LET b$(xx,yyy)="⊠": PRINT I
NK 5;AT 3+xx,3+yyy;b$(xx,yyy)
 660 IF yy=0 OR xxx=11 THEN GO T
O 680
 670 LET b$(xxx,yy)="⊠": PRINT I
NK 5;AT 3+xxx,3+yy;b$(xxx,yy)
 680 IF xxx=11 OR yyy=11 THEN GO
 TO 700
 690 LET b$(xxx,yyy)="⊠": PRINT
INK 5;AT 3+xxx,3+yyy;b$(xxx,yyy)
 700 IF xx=0 THEN GO TO 720
 710 IF a$(xx,nn)="█" THEN GO TO
900
 720 IF yy=0 THEN GO TO 740
 730 IF a$(no,yy)="█" THEN GO TO
 1200
 740 IF xxx=11 THEN GO TO 760
 750 IF a$(xxx,nn)="█" THEN GO T
O 900
 760 IF yyy=11 THEN GO TO 780
 770 IF a$(no,yyy)="█" THEN GO T
O 1200
 780 IF xx=0 THEN GO TO 800
 790 LET b$(xx,nn)="⊠": PRINT IN
K 5;AT xx+3,nn+3;b$(xx,nn)
 800 IF yy=0 THEN GO TO 820
 810 LET b$(no,yy)="⊠": PRINT IN
K 5;AT no+3,yy+3;b$(no,yy)
 820 IF xxx=11 THEN GO TO 840
 830 LET b$(xxx,nn)="⊠": PRINT I
NK 5;AT xxx+3,nn+3;b$(xxx,nn)
 840 IF yyy=11 THEN GO TO 3000
 850 LET b$(no,yyy)="⊠": PRINT I
NK 5;AT no+3,yyy+3;b$(no,yyy)
 860 GO TO 3000
 890 REM verticales
 900 IF xx=0 THEN GO TO 980
 910 IF a$(xx,nn)="□" THEN GO TO
 980
 920 IF xx-1=0 THEN GO TO 990
 930 IF a$(xx-1,nn)="□" THEN GO
TO 990
 940 IF xx-2=0 THEN GO TO 1000
 950 IF a$(xx-2,nn)="□" THEN GO
TO 1000
 970 LET d=xx-2: LET dd=no: GO T
O 1110
 980 LET d=no: GO TO 1010
 990 LET d=xx: GO TO 1010
1000 LET d=xx-1
1010 IF xxx=11 THEN GO TO 1080
1020 IF a$(xxx,nn)="□" THEN GO T
O 1080
1030 IF xxx+1=11 THEN GO TO 1090
1040 IF a$(xxx+1,nn)="□" THEN GO
 TO 1090
1050 IF xxx+2=11 THEN GO TO 1100
1060 IF a$(xxx+2,nn)="□" THEN GO
 TO 1100
1070 LET dd=xxx+2: GO TO 1110
1080 LET dd=no: GO TO 1110
1090 LET dd=xxx: GO TO 1110
1100 LET dd=xxx+1
1110 FOR n=d TO dd
1120 IF b$(n,nn)="□" THEN GO TO
3000
1130 NEXT n
1140 IF d-1=0 THEN GO TO 1160
1150 LET b$(d-1,nn)="⊠": PRINT I
NK 5;AT d+2,nn+3;b$(d-1,nn)
1160 IF dd+1=11 THEN GO TO 3000
1170 LET b$(dd+1,nn)="⊠": PRINT
INK 5;AT dd+4,nn+3;b$(dd+1,nn)
1180 GO TO 3000
1190 REM horizontales
1200 IF yy=0 THEN GO TO 1280
1210 IF a$(no,yy)="□" THEN GO TO
 1280
1220 IF yy-1=0 THEN GO TO 1290
1230 IF a$(no,yy-1)="□" THEN GO
TO 1290
1240 IF yy-2=0 THEN GO TO 1300
1250 IF a$(no,yy-2)="□" THEN GO
TO 1300
1270 LET d=yy-2: LET dd=nn: GO T
O 1310
1280 LET d=nn: GO TO 1310
1290 LET d=yy: GO TO 1310
1300 LET d=yy-1
1310 IF yyy=11 THEN GO TO 1380
1320 IF a$(no,yyy)="□" THEN GO T
O 1380
1330 IF yyy+1=11 THEN GO TO 1390
1340 IF a$(no,yyy+1)="□" THEN GO
 TO 1390
1350 IF yyy+2=11 THEN GO TO 1400
1360 IF a$(no,yyy+2)="□" THEN GO
 TO 1400
1370 LET dd=yyy+2: GO TO 1410
1380 LET dd=nn: GO TO 1410
1390 LET dd=yyy: GO TO 1410
1400 LET dd=yyy+1: GO TO 1410
1410 FOR n=d TO dd
1420 IF b$(no,n)="□" THEN GO TO
3000
1430 NEXT n
1440 IF d-1=0 THEN GO TO 1460
1450 LET b$(no,d-1)="⊠": PRINT I
NK 5;AT no+3,d+2;b$(no,d-1)
1460 IF dd+1=11 THEN GO TO 3000
1470 LET b$(no,dd+1)="⊠": PRINT
INK 5;AT no+3,dd+4;b$(no,dd+1)
3010 LET i=INT (RND*27+1)
3020 IF i=1 THEN GO SUB 5040
3021 IF i=2 THEN GO SUB 5041
3022 IF i=3 THEN GO SUB 5042
3023 IF i=4 THEN GO SUB 5043
3024 IF i=5 THEN GO SUB 5044
3025 IF i=6 THEN GO SUB 5045
3026 IF i=7 THEN GO SUB 5046
3027 IF i=8 THEN GO SUB 5047
3028 IF i=9 THEN GO SUB 5048
3029 IF i=10 THEN GO SUB 5049
3030 IF i=11 THEN GO SUB 5050
3031 IF i=12 THEN GO SUB 5051
3032 IF i=13 THEN GO SUB 5052
3033 IF i=14 THEN GO SUB 5053
3034 IF i=15 THEN GO SUB 5054
3035 IF i=16 THEN GO SUB 5055
3036 IF i=17 THEN GO SUB 5056
3037 IF i=18 THEN GO SUB 5057
3038 IF i=19 THEN GO SUB 5058
3039 IF i=20 THEN GO SUB 5059
3040 IF i=21 THEN GO SUB 5060
3041 IF i=22 THEN GO SUB 5061
3042 IF i=23 THEN GO SUB 5062
3043 IF i=24 THEN GO SUB 5063
3044 IF i=25 THEN GO SUB 5064
3045 IF i=26 THEN GO SUB 5065
3046 IF i=27 THEN GO SUB 5066
3050 PRINT INK 1;AT 16,2;r$(v);A
T 18,2;r$(vv)
3060 FOR n=1 TO 4: BEEP .08,1: B
EEP .08,7: NEXT n
3070 GO TO 420
3500 PRINT AT 0,0;"
                ";AT 1,0;"
   ABANDONASTE              ": GO
 TO 4020
4001 LET v=19: LET vv=20: PRINT
AT 18,2;r$(v);AT 16,2;r$(vv)
4010 PRINT AT 0,0;"
                ";AT 1,0;"
 RESULTADO FINAL              "
4025 FOR n=1 TO 10: FOR m=1 TO 1
0: INK 1: BEEP .03,8: PRINT FLAS
H 1;AT n+3,m+3;a$(n,m): NEXT m:
NEXT n
4030 FOR n=1 TO 5: BEEP .07,7: B
EEP .09,9: BEEP .015,7: BEEP .06
,2: BEEP .09,7: NEXT n
4040 INK 4: PRINT FLASH 1;AT 16,
2;"   PRESIONA UNA TECLA     ";A
T 18,2;"     TE HAS DIVERTIDO ?
  ": PAUSE 0: BEEP .07,7
4045 PAPER 5
4050 CLS : PRINT INK 3;AT 1,1;"B
ARCOS": BEEP .05,5: PRINT INK 4;
AT 1,7;"BARCOS": BEEP .05,5: PRI
NT INK 2;AT 1,13;"BARCOS": BEEP
.05,5: PRINT INK 7;AT 1,19;"BARC
OS": BEEP .05,5: PRINT INK 1;AT
1,25;"BARCOS": BEEP .05,5
4060 INK 2: PRINT FLASH 1;AT 5,3
;"1";AT 7,3;"2";AT 15,3;"  CUAL
QUIER OTRA TECLA    "
4070 PRINT INK 3;AT 5,5;"GRABAR
EL JUEGO";AT 7,5;"DESTRUIR EL PR
OGRAMA";AT 12,3;"   JUGAR A LOS
BARCOS": PAUSE 0
4080 LET j$=INKEY$: BEEP .07,7
4090 IF j$="1" THEN GO TO 4500
4100 IF j$="2" THEN GO TO 4400
4110 GO TO 14
4400 PAPER 1: CLS
4410 INK 6: PRINT FLASH 1;AT 10,
0;" ESTE PROGRAMA SE AUTODESTRUI
RA ";AT 11,0;"
                ";AT 9,0;"
                        ";AT 13,
0;"
   ";AT 12,0;"      EN UNOS 25 S
EGUNDOS       "
4415 FOR n=1 TO 20: BEEP .6,7: B
EEP .6,5: NEXT n
4420 NEW
4500 CLS : PRINT AT 10,3;" DESCO
NECTE ## EAR ##  Y";AT 12,3;" PU
LSE CUALQUIER TECLA ": PAUSE 0
4510 SAVE "BARCOS" LINE 9999
4520 CLS : PRINT AT 10,1;" CONEC
TE ## EAR ## QUE VAMOS";AT 12,2;
" A VERIFICAR LA GRABACION ";AT
14,1;" REBOBINE LA CINTA , PULSE
";AT 16,3;" PLAY Y CUALQUIER TEC
LA": PAUSE 0
4530 VERIFY "BARCOS"
4550 CLS : PRINT AT 10,8;"  ES C
ORRECTA": PAUSE 60: GO TO 4045
5000 LET v=1: LET vv=10: RETURN
5001 LET v=1: LET vv=3: RETURN
5002 LET v=1: LET vv=8: RETURN
5003 LET v=1: LET vv=9: RETURN
5004 LET v=1: LET vv=12: RETURN
5005 LET v=1: LET vv=15: RETURN
5006 LET v=1: LET vv=18: RETURN
5007 LET v=1: LET vv=20: RETURN
5010 LET v=2: LET vv=20: RETURN
5011 LET v=2: LET vv=7: RETURN
5012 LET v=2: LET vv=9: RETURN
5013 LET v=2: LET vv=10: RETURN
5014 LET v=5: LET vv=6: RETURN
5015 LET v=5: LET vv=8: RETURN
5016 LET v=5: LET vv=9: RETURN
5017 LET v=5: LET vv=10: RETURN
5018 LET v=5: LET vv=12: RETURN
5019 LET v=5: LET vv=15: RETURN
5020 LET v=5: LET vv=18: RETURN
5021 LET v=6: LET vv=8: RETURN
5022 LET v=6: LET vv=5: RETURN
5023 LET v=6: LET vv=10: RETURN
5024 LET v=6: LET vv=20: RETURN
5025 LET v=7: LET vv=20: RETURN
5026 LET v=7: LET vv=9: RETURN
5027 LET v=8: LET vv=6: RETURN
5028 LET v=4: LET vv=6: RETURN
5029 LET v=4: LET vv=8: RETURN
5030 LET v=4: LET vv=9: RETURN
5031 LET v=4: LET vv=12: RETURN
5032 LET v=4: LET vv=14: RETURN
5033 LET v=4: LET vv=15: RETURN
5034 LET v=4: LET vv=18: RETURN
5035 LET v=9: LET vv=6: RETURN
5036 LET v=9: LET vv=2: RETURN
5040 LET v=11: LET vv=12: RETURN
5041 LET v=11: LET vv=14: RETURN
5042 LET v=11: LET vv=16: RETURN
5043 LET v=11: LET vv=18: RETURN
5044 LET v=13: LET vv=11: RETURN
5045 LET v=13: LET vv=14: RETURN
5046 LET v=13: LET vv=15: RETURN
5047 LET v=13: LET vv=16: RETURN
5048 LET v=13: LET vv=18: RETURN
5049 LET v=14: LET vv=11: RETURN
5050 LET v=14: LET vv=12: RETURN
5051 LET v=14: LET vv=15: RETURN
5052 LET v=14: LET vv=16: RETURN
5053 LET v=14: LET vv=17: RETURN
5054 LET v=14: LET vv=18: RETURN
5055 LET v=15: LET vv=11: RETURN
5056 LET v=16: LET vv=20: RETURN
5057 LET v=16: LET vv=12: RETURN
5058 LET v=16: LET vv=14: RETURN
5059 LET v=16: LET vv=18: RETURN
5060 LET v=17: LET vv=11: RETURN
5061 LET v=17: LET vv=12: RETURN
5062 LET v=17: LET vv=14: RETURN
5063 LET v=17: LET vv=15: RETURN
5064 LET v=17: LET vv=16: RETURN
5065 LET v=17: LET vv=17: RETURN
5066 LET v=17: LET vv=18: RETURN
7980 REM GRAFICOS
7990 DATA 255,BIN 10000001,BIN 1
0000001,BIN 10000001,BIN 1000000
1,BIN 10000001,BIN 10000001,255
7999 DATA 255,BIN 11000011,BIN 1
0100101,BIN 10011001,BIN 1001100
1,BIN 10100101,BIN 11000011,255
8000 RESTORE : FOR n=0 TO 7
8010 BEEP .07,7: BORDER n: READ
a: POKE USR "□"+n,a
8020 POKE USR "█"+n,255
8030 NEXT n
8035 FOR n=0 TO 7: BEEP .07,7: B
ORDER n: READ b: POKE USR "⊠"+n,
b: NEXT n: BORDER 2
8036 GO TO 8050
8040 FOR n=1 TO 10: FOR m=1 TO 1
0: LET a$(n,m)="□": LET b$(n,m)=
"□": NEXT m: NEXT n
8041 RETURN
8050 LET r$(20)="
                "
8051 LET r$(1)="DISPARADO ANTERI
ORMENTE       "
8052 LET r$(2)="AGUA DE MANAGUA,
 QUE RICA     "
8053 LET r$(3)="COMENZAMOS YA ?
              "
8054 LET r$(4)="LE HAS DADO A UN
 TIBURON"
8055 LET r$(5)="CASI CASI ...PER
O NO          "
8056 LET r$(6)="NO DAS NI UNA
              "
8057 LET r$(7)="QUE NO !! QUE NO
 !!           "
8058 LET r$(8)="OYE ME ABURRES
              "
8059 LET r$(9)="VAS DE CRANEO
              "
8060 LET r$(10)="LO TIENES CLARO
              "
8061 LET r$(11)="GUAY DEL PARAGU
AY            "
8062 LET r$(12)="JA!! JA!! JA!!
JE!! JE!!     "
8063 LET r$(13)="DIANA DE PURA S
UERTE         "
8064 LET r$(14)="QUE PUNTERIA !
              "
8065 LET r$(15)="TE MEJORAS ?
              "
8066 LET r$(16)="MARINERITOS AL
AGUA          "
8067 LET r$(17)="LE DIO!! LE DIO
!!            "
8068 LET r$(18)="COMO LO HACES?
              "
8069 LET r$(19)="HUNDISTE LA FLO
TA            "
8100 RETURN
9999 RUN
```






**Aventura cinematográfica**

# GHOSTBUSTERS

**Ha llegado de Inglaterra un nuevo programa para Spectrum, GHOSTBUSTERS, basado en una de las películas más taquilleras del momento. Ofrecemos en primicia un análisis del juego para nuestros lectores.**

*Activision/ERBE*
*48 K*
*Tipo de juego: Arcade*
*PVP: 2.700*

No es la primera vez, y desde luego estamos seguros que tampoco será la última, que en una película o un conocido libro sirvan de inspiración en la creación de un programa de juegos para ordenador.

Basado en una conocida película con gran éxito taquillero en la actualidad, Ghostbusters (Los Cazafantasmas), nos convierte en los protagonistas del film, de modo que tenemos que ir siguiendo los mismos pasos que daban éstos en la película, con el fin de evitar que la energía de los fantasmas acabe destruyendo la ciudad.

Para aquellos que no hayan visto la película, vamos a tratar de resumirla un poco: se trata de tres estudiantes universitarios que trabajan en el departamento de investigación, y dedican su tiempo a controlar los efectos producidos por ciertos fenómenos paranormales, provocados por fantasmas que deambulan por la ciudad. Expulsados del centro, por considerar sus actividades como una pérdida de tiempo, comienzan una caza de fantasmas por la ciudad donde los acontecimientos se van desarrollando con demasiada rapidez.

Como en la película, existen dos personajes que van a resultar claves a lo largo del juego. Los dos han sido poseídos por las fuerzas del mal y van a representar dos símbolos, que juntos, pueden resultar muy peligrosos. Son el guardián de la puerta, que en la película era una mujer poseída, y el señor de la llave. Cuando ambos se encuentren juntos en el templo de Zuul, el final es inevitable. En el juego están representados por una llave y una cerradura que se mueven aleatoriamente por las calles de la ciudad.

El templo de Zuul es el lugar de reunión de todos los fantasmas de la ciudad, en nuestro juego se encuentra situado en el centro de un mapa de las calles, al cual podremos acceder en las diferentes fases del mismo, para observar la posición en la que nos hallamos y ver dónde se encuentran nuestros enemigos.

Al comienzo del juego tenemos que elegir, como ocurría en la película, un medio de transporte para realizar nuestra misión. Podemos hacerlo entre cuatro diferentes, pero siempre teniendo en cuenta que disponemos de una cantidad de dinero limitada y que cada uno de los vehículos tiene un precio y unas características diferentes. Nuestro dinero se irá acabando a medida que vayamos comprando cosas y hay que tener en cuenta que además del coche, necesitamos comprar el equipo para cazar fantasmas. Hay tres tipos de material: uno, motorizado que va a consumirnos energía de las pilas que llevamos a la espalda; otro, de captura, el cual contiene diferentes trampas, de las cuales al menos una, vamos a necesitar si queremos atrapar a alguno de nuestros enemigos, y, finalmente, el equipo de almacenaje que nos permite confinar a los fantasmas atrapados en un lugar seguro, donde no puedan hacer daño a nadie.

El Detector de Energía PK, es una pieza importantísima que nos permite localizar rápidamente la posición de un fantasma.

El Intensificador de Imagen, nos ayuda a ver mucho mejor a nuestros enemigos, haciéndonos más fácil la captura.

Una vez hemos seleccionado el equipo, podremos acceder a un mapa de calles, donde se nos muestra la situación en general. Nosotros tenemos que guiar el vehículo hacia el edificio que parpadee, ya que esto es la señal de la presencia de un fantasma.

Cada vez que un fantasma consiga llegar al templo de Zuul, el valor del PK aumentará en 100 unidades.

Cuando hemos localizado a un fantasma, tenemos que tratar de acorralarlo con nuestras unidades ionizadoras, formando un campo energético, dentro del cual dispararemos a nuestro enemigo. Cuando acertemos, éste caerá atrapado en nuestras redes. Por cada uno que atrapemos aumentará nuestra cuenta.

Si los fantasmas logran unirse y formar un hombre Marsahmallow, éste, al igual que ocurría en la película, intentará destruir todos los edificios que encuentra a su paso. Nuestra misión, si esto ocurre, es tratar de evitarlo, poniéndole trampas a lo largo de su recorrido.

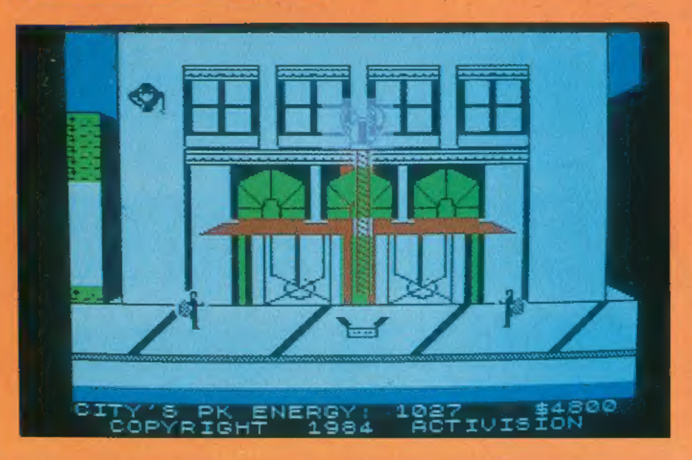
El juego es original en su concepción y ha aprovechado el éxito de la película para tratar de crear un clima atractivo para el usuario de Spectrum. Las instrucciones, que vienen en castellano, explican de una forma muy completa cómo hay que jugar, aunque de todas formas, esto no resultará muy difícil para aquellos que ya hayan visto el film.

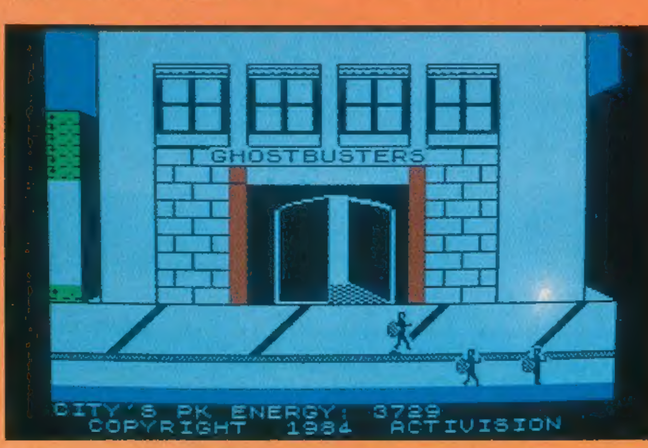
Los gráficos, que están construidos conforme a la idea del juego, resultan muy propicios en todas las fases del mismo. El movimiento es muy simple en su manejo y ofrece, además, la posibilidad de que sea utilizado con Joystick.

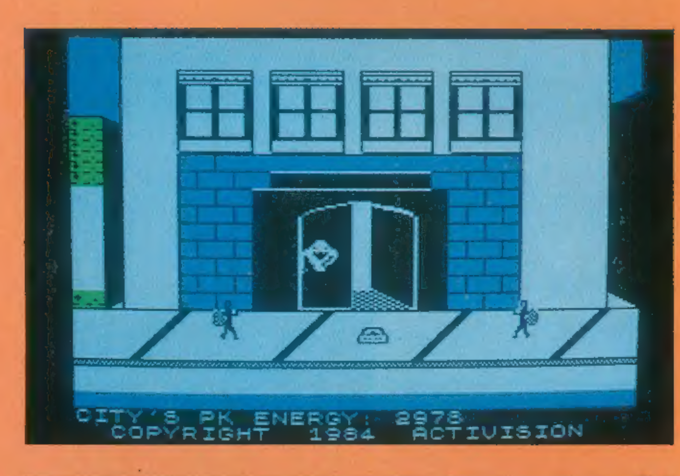
Es, en definitiva, un buen juego, original en su tratamiento y que inicia quizás una nueva concepción en cuanto a los programas de juegos que a partir de ahora podrían empezar a basarse en conocidas películas, convirtiendo, de este modo, a nuestro Spectrum, en un nuevo Hollywood computerizado.

La música que escuchamos al principio es también la del conocido tema «Ghostbusters» que tanto suena en las emisoras de radio.



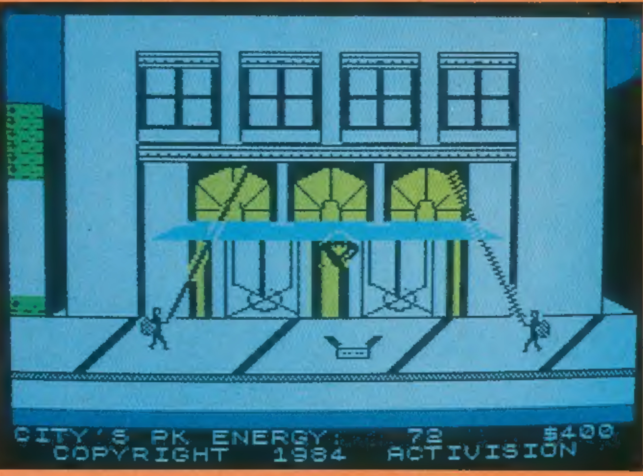


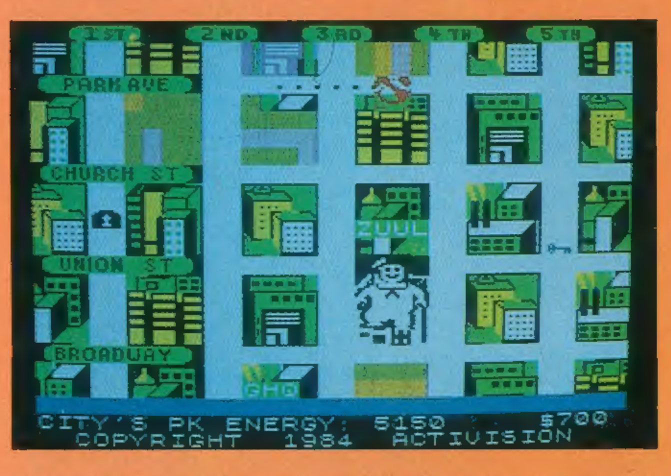



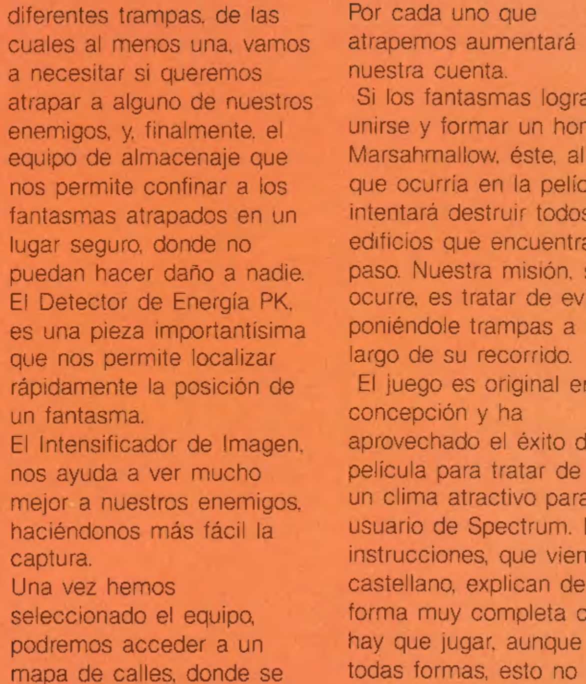



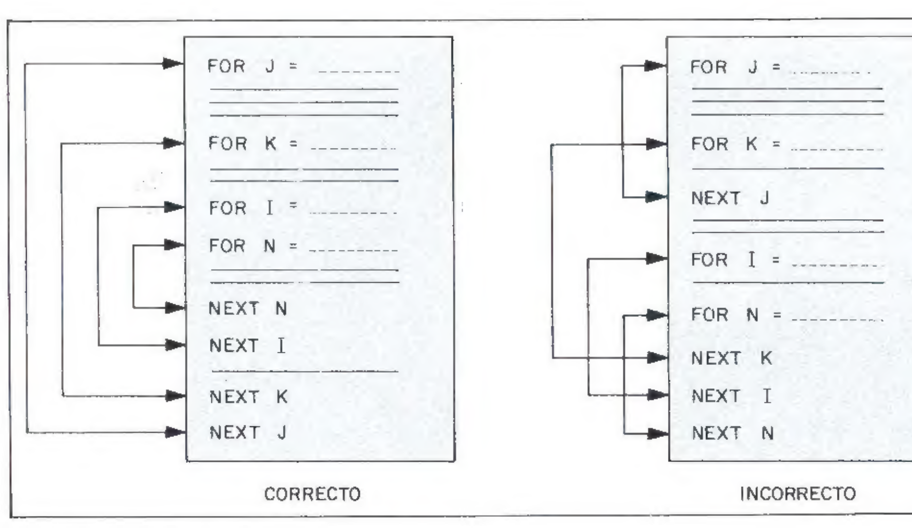









**Anidamiento de bucles.**

Cuando «STEP» se omite, el intérprete BASIC toma por defecto el valor «1», como se ha visto en los ejemplos anteriores, que la variable de control asumía inicialmente el valor del límite inferior y se incrementaba en uno hasta alcanzar el del límite superior.

Su estructura es la siguiente:

| SENTENCIA | ARGUMENTO |
|---|---|
| FOR | ... TO STEP expresión |

Ejemplo:

FOR j = 8 TO 32 STEP 4

la variable de control «j» tomará el valor «8» y se incrementará de cuatro en cuatro hasta alcanzar el valor «32», es decir, 8-12-16-20 ... 32. El bucle se realiza siete veces.

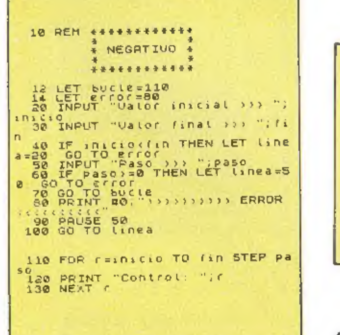
Ejemplo:

```
10 REM *********
       *       *
       * STEP  *
       *       *
       *********
20 INPUT "Paso >>> ";paso
30 FOR n=0 TO 100 STEP paso
40 PRINT "Variable de control:
";n
50 NEXT n
```

el número de veces que se ejecuta el programa anterior, así como los valores que toma la variable de control «n», depende del que se asigne a la variable «paso», introducida por el teclado.

El *paso* puede ser una expresión negativa, en este caso, el valor inicial debe ser superior al final.

Ejemplo:



```
10 REM *************
       *           *
       * NEGATIVO  *
       *           *
       *************
12 LET bucle=110
14 LET error=80
20 INPUT "Valor inicial >>> ";
inicio
30 INPUT "Valor final >>> ";fi
n
40 IF inicio<fin THEN LET line
a=20: GO TO error
50 INPUT "Paso >>> ";paso
60 IF paso>=0 THEN LET linea=5
0: GO TO error
70 GO TO bucle
80 PRINT #0;">>>>>>>>>> ERROR
<<<<<<<<<<"
90 PAUSE 50
100 GO TO linea
110 FOR r=inicio TO fin STEP pa
so
120 PRINT "Control: ";r
130 NEXT r
```

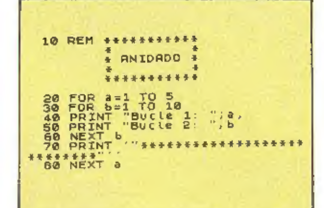
este programa sólo se ejecuta con bucles decrecientes, ya que presenta un mensaje de ERROR si el valor inicial es inferior al final o si el paso es mayor o igual a cero.

## Bucles anidados

Se denomina bucle *anidado*, aquel que contiene otro en su interior; no existe límite en cuanto a la cantidad de bucles que se pueden anidar.

Los bucles deben estar completamente *encajados*, por lo que no deben solaparse.

Ejemplo:



```
10 REM ************
       *          *
       * ANIDADO  *
       *          *
       ************
20 FOR a=1 TO 5
30 FOR b=1 TO 10
40 PRINT "Bucle 1: ";a,
50 PRINT "Bucle 2: ";b
60 NEXT b
70 PRINT '"**************************
********"''
80 NEXT a
```

El bucle interior «b» se ejecuta 5Ø veces aunque sus límites van del 1 al 1Ø, ya se multiplica por los del exterior «a». El programa visualiza los valores de las dos variables de



**PROGRAMA 2**

```
10 REM ***************
       *             *
       * CURSO BASIC *
       *             *
       ***************
       *             *
       *   AREAS Y   *
       *             *
       * PERIMETROS  *
       *             *
       ***************
20 BORDER 1: PAPER 1: INK 7: C
LS
30 REM ************
       *          *
       * OPCIONES *
       *          *
       ************
40 PRINT AT 1,0;"CALCULO DE AR
EAS   © MICROHOBBY"
50 PRINT AT 4,9;"MENU DE OPCIO
NES"
60 PRINT AT 5,7;"_______________
"
70 PRINT AT 8,8;"1 - TRIANGULO
"
80 PRINT AT 10,8;"2 - RECTANGU
LO"
90 PRINT AT 12,8;"3 - CUADRADO
"
100 PRINT AT 14,8;"4 - ROMBO"
110 PRINT AT 16,8;"5 - ROMBOIDE
"
120 PRINT AT 18,8;"6 - TRAPECIO
"
130 PRINT AT 20,8;"7 - CIRCUNFE
RENCIA"
140 INPUT "Introduzca la opcion
 deseada  >>> ";opcion
150 REM ****************
        *              *
        * VERIFICACION *
        *              *
        ****************
160 IF opcion<1 OR opcion>7 THE
N GO TO 140
165 CLS
170 IF opcion=1 THEN GO TO 250
180 IF opcion=2 THEN GO TO 450
190 IF opcion=3 THEN GO TO 700
200 IF opcion=4 THEN GO TO 970
210 IF opcion=5 THEN GO TO 1170
220 IF opcion=6 THEN GO TO 1380
230 IF opcion=7 THEN GO TO 1630
240 GO TO 40
250 REM *************
        *           *
        * TRIANGULO *
        *           *
        *************
260 PRINT AT 2,11;"TRIANGULO"
270 PRINT AT 3,11;"_________"
280 PLOT INK 4;16,47
290 DRAW INK 4;63,0
300 DRAW INK 4;-32,73
310 DRAW INK 4;-31,-73
312 PRINT AT 17,5;"b"
314 PRINT AT 12,5;"h"
320 REM *****************
        *               *
        * ENTRADA DATOS *
        *               *
        *****************
330 INPUT "Base >>> ";base
340 PRINT AT 7,11;"b : ";base
350 INPUT "Altura >>> ";altura
360 PRINT AT 9,11;"h : ";altura
370 REM ************
        *          *
        * CALCULOS *
        *          *
        ************
380 LET area=base*altura/2
390 PRINT AT 12,11;"AREA      :
";area
400 LET lado=SQR (altura↑2+(bas
e/2)↑2)
410 LET perimetro=lado*2+base
430 PRINT AT 14,11;"PERIMETRO:
";perimetro
440 GO TO 9000
450 REM **************
        *            *
        * RECTANGULO *
        *            *
        **************
460 PRINT AT 2,11;"RECTANGULO"
470 PRINT AT 3,11;"__________"
480 PLOT INK 4;16,47
490 DRAW INK 4;63,0
500 DRAW INK 4;0,80
510 DRAW INK 4;-63,0
520 DRAW INK 4;0,-80
530 PRINT AT 17,5;"b"
540 PRINT AT 10,0;"h"
550 REM *****************
        *               *
        * ENTRADA DATOS *
        *               *
        *****************
560 INPUT "Base >>> ";base
570 PRINT AT 7,13;"b : ";base
580 INPUT "Altura >>> ";altura
590 PRINT AT 9,13;"h : ";altura
600 REM ************
        *          *
        * CALCULOS *
        *          *
        ************
610 LET area=base*altura
620 PRINT AT 12,13;"AREA      :
";area
630 LET perimetro=base*2+altura
*2
640 PRINT AT 14,13;"PERIMETRO:
";perimetro
650 GO TO 9000
700 REM ************
        *          *
        * CUADRADO *
        *          *
        ************
710 PRINT AT 2,13;"CUADRADO"
720 PRINT AT 3,13;"________"
730 PLOT INK 4;16,47
740 DRAW INK 4;63,0
750 DRAW INK 4;0,63
760 DRAW INK 4;-63,0
770 DRAW INK 4;0,-63
780 PRINT AT 17,5;"l"
790 REM *****************
        *               *
        * ENTRADA DATOS *
        *               *
        *****************
800 INPUT "Lado >>> ";lado
900 PRINT AT 7,13;"l : ";lado
910 REM ************
        *          *
        * CALCULOS *
        *          *
        ************
920 LET area=lado↑2
930 PRINT AT 12,13;"AREA      :
";area
940 LET perimetro=lado*4
950 PRINT AT 14,13;"PERIMETRO:
";perimetro
960 GO TO 9000
970 REM
```

```
          *********
          *       *
          * ROMBO *
          *       *
          *********
 980 PRINT AT 2,14;"ROMBO"
 990 PRINT AT 3,14;"_____"
1000 PLOT INK 4;40,40
1010 DRAW INK 4;24,48
1020 DRAW INK 4;-24,48
1030 DRAW INK 4;-24,-48
1040 DRAW INK 4;24,-48
1050 REM
         *****************
         *               *
         * ENTRADA DATOS *
         *               *
         *****************
1060 INPUT "Diagonal mayor >>> ";mayor
1070 PRINT AT 7,12;"D : ";mayor
1080 INPUT "Diagonal menor >>> ";menor
1090 PRINT AT 9,12;"d : ";menor
1100 REM
         ***********
         *         *
         * CALCULOS *
         *         *
         ***********
1110 LET area=mayor*menor/2
1120 PRINT AT 12,12;"AREA     : ";area
1130 LET lado=SQR ((mayor/2)↑2+(menor/2)↑2)
1140 LET perimetro=lado*4
1150 PRINT AT 14,12;"PERIMETRO: ";perimetro
1160 GO TO 9000
1170 REM
         ************
         *          *
         * ROMBOIDE *
         *          *
         ************
1180 PRINT AT 2,12;"ROMBOIDE"
1190 PRINT AT 3,12;"________"
1200 PLOT INK 4;8,47
1210 DRAW INK 4;64,0
1220 DRAW INK 4;16,40
1230 DRAW INK 4;-64,0
1240 DRAW INK 4;-16,-40
1250 PRINT AT 17,5;"b"
1260 PRINT AT 13,4;"h"
1265 PRINT AT 13,0;"l"
1270 REM
         *****************
         *               *
         * ENTRADA DATOS *
         *               *
         *****************
1280 INPUT "Base >>> ";base
1290 PRINT AT 7,13;"b : ";base
1300 INPUT "Altura >>> ";altura
1310 PRINT AT 9,13;"h : ";altura
1312 INPUT "Lado >>> ";lado
1314 PRINT AT 11,13;"l : ";lado
1320 REM
         ***********
         *         *
         * CALCULOS *
         *         *
         ***********
1330 LET area=base*altura
1340 PRINT AT 14,13;"AREA     : ";area
1350 LET perimetro=lado*2+base*2
1360 PRINT AT 16,13;"PERIMETRO: ";perimetro
1370 GO TO 9000
1380 REM
         ***********
         *         *
         * TRAPECIO *
         *         *
         ***********
1390 PRINT AT 2,12;"TRAPECIO"
1400 PRINT AT 3,12;"________"
1410 PLOT INK 4;8,47
1420 DRAW INK 4;80,0
1430 DRAW INK 4;-16,64
1440 DRAW INK 4;-48,0
1450 DRAW INK 4;-16,-64
1460 PRINT AT 17,5;"B"
1470 PRINT AT 12,3;"h"
1480 PRINT AT 6,6;"b"
1490 REM
         *****************
         *               *
         * ENTRADA DATOS *
         *               *
         *****************
1500 INPUT "Base mayor >>> ";mayor
1510 PRINT AT 7,13;"B : ";mayor
1520 INPUT "Base menor >>> ";menor
1530 PRINT AT 9,13;"b : ";menor
1540 INPUT "Altura >>> ";altura
1550 PRINT AT 11,13;"h : ";altura
1560 REM
         ***********
         *         *
         * CALCULOS *
         *         *
         ***********
1570 LET area=(mayor+menor)*altura/2
1580 PRINT AT 14,13;"AREA     : ";area
1590 LET lado=SQR (((mayor-menor)/2)↑2+altura↑2)
1600 LET perimetro=mayor+menor+lado*2
1610 PRINT AT 16,13;"PERIMETRO: ";perimetro
1620 GO TO 9000
1630 REM
         *****************
         *               *
         * CIRCUNFERENCIA *
         *               *
         *****************
1640 PRINT AT 2,9;"CIRCUNFERENCIA"
1650 PRINT AT 3,9;"______________"
1660 CIRCLE INK 4;40,88,32
1670 PRINT AT 9,3;"r"
1680 REM
         *****************
         *               *
         * ENTRADA DATOS *
         *               *
         *****************
1690 INPUT "Radio >>> ";radio
1700 PRINT AT 7,13;"r : ";radio
1710 REM
         ***********
         *         *
         * CALCULOS *
         *         *
         ***********
1720 LET area=PI*radio↑2
1730 PRINT AT 10,12;"AREA     : ";area
1740 LET perimetro=2*PI*radio
1750 PRINT AT 12,12;"PERIMETRO: ";perimetro
1760 GO TO 9000
9000 REM
         *****************
         *               *
         * MAS CALCULOS? *
         *               *
         *****************
9010 INPUT "Quiere continuar (S o N) >>> "; LINE r$
9020 IF r$="n" OR r$="N" THEN STOP
9030 IF r$="s" OR r$="S" THEN CLS : GO TO 40
9040 GO TO 9010
```



## Programa

Como programa de repaso de las sentencias «GO TO» e «IF ... THEN ...», se propone el programa número «2». Sálvelo, por ejemplo, de la forma:

SAVE "AREAS" LINE 1Ø

Este programa calcula el *área* y el *perímetro* de las siguientes figuras geométricas.

- — TRIANGULO.
- — RECTANGULO.
- — CUADRADO.
- — ROMBO.
- — ROMBOIDE.
- — TRAPECIO.
- — CIRCULO.

El programa, al autoejecutarse, presenta en pantalla un menú con las diversas opciones; seleccionando una de ellas, pasaremos a una pantalla en la que aparecerá dibujada la figura geométrica correspondiente, ésta es realizada con ayuda de las sentencias «PLOT», «DRAW» o «CIRCLE». Una vez introducidos los datos (lado, altura,...) que nos pide el ordenador, los resultados correspondientes al «área» y al «perímetro» serán visualizados.

Para retornar al menú principal debe pulsar «s» o «S», pulsando «n» o «N» el programa se detendrá presentando el mensaje:

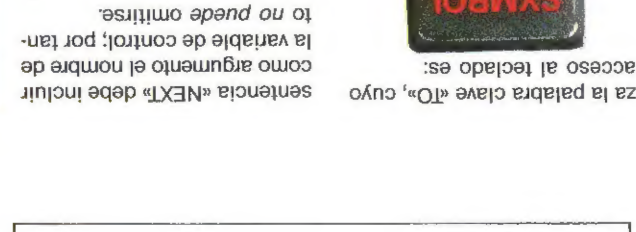
9 STOP statement, 9Ø 2Ø :2

La estructura del programa es:

| | | |
|---|---|---|
| 1Ø: | | Comentario con el nombre del programa. |
| 2Ø: | | Asignación de los colores de la pantalla. |
| 3Ø-13Ø | : | Menú de opciones. |
| 14Ø | : | Entrada de «opción». |
| 15Ø-24Ø | : | Verificación y selección. |
| 25Ø-314 | : | Dibujo de *triángulo*. |
| 32Ø-36Ø | : | Entrada de datos, (base y altura). |
| 37Ø-43Ø | : | Cálculo y visualización. |
| 44Ø | : | Salto de la rutina «FIN». |
| 45Ø-54Ø | : | Dibujo del *rectángulo*. |
| 55Ø—59Ø | : | Entrada de datos, (base y altura). |
| 6ØØ-64Ø | : | Cálculo y visualización. |
| 65Ø | : | Salto a la rutina «FIN». |
| 7ØØ-78Ø | : | Dibujo del *cuadrado*. |
| 79Ø-9ØØ | : | Entrada de datos, (lado). |
| 91Ø-95Ø | : | Cálculo y visualización. |
| 96Ø | : | Salto a la rutina «FIN». |
| 97Ø-1Ø4Ø | : | Dibujo del *rombo*. |
| 1Ø5Ø-1Ø9Ø | : | Entrada de datos, (diagonal mayor y diagonal menor). |
| 11ØØ-115Ø | : | Cálculo y visualización. |
| 116Ø | : | Salto a la rutina «FIN». |
| 117Ø-1265 | : | Dibujo del *romboide*. |
| 127Ø-1314 | : | Entrada de datos, (base, altura y lado). |
| 132Ø-136Ø | : | Cálculo y visualización. |
| 137Ø | : | Salto a la rutina «FIN». |
| 138Ø-148Ø | : | Dibujo del *trapecio*. |
| 149Ø-155Ø | : | Entrada de datos, (base mayor, base menor y altura). |
| 156Ø-161Ø | : | Cálculo y visualización. |
| 162Ø | : | Salto a la rutina «FIN». |
| 163Ø-167Ø | : | Dibujo del *círculo*. |
| 168Ø-17ØØ | : | Entrada de datos, (radio). |
| 171Ø-175Ø | : | Cálculo y visualización. |
| 176Ø | : | Salto a la rutina «FIN». |
| 9ØØØ-9Ø4Ø | : | Rutina «FIN». |



za la palabra clave «TO», cuyo acceso al teclado es:




La estructura de «NEXT» tiene el siguiente formato:

| SENTENCIA | ARGUMENTO |
|---|---|
| NEXT | variable |

Ejemplos:

1Ø FOR j = 2 TO 1Ø

5Ø NEXT j

En palabras sencillas, la explicación del ejemplo anterior es:

Ejecutar las instrucciones siguientes al «FOR ... TO ...» hasta que la variable «j», partiendo de «2», tome el valor «1Ø»; la sentencia «NEXT» se encarga de incrementar el valor de esta variable; el bucle se repite por tanto, nueve veces.

La variable de control del bucle, sólo puede estar formada por una letra, si existe otra variable con el mismo nombre, ésta es borrada y asume el nuevo valor. La expresión anterior al «TO» es el valor inicial que debe tomar la variable de control, y la expresión posterior el valor final. A diferencia de otro lenguajes BASIC, la sentencia «NEXT» debe incluir como argumento el nombre de la variable de control; por tanto *no puede* omitirse.

Ejemplo:

```
10 REM **************
       *            *
       * FOR...NEXT *
       *            *
       **************
20 FOR n=1 TO 61
30 PRINT "MICROHOBBY ";
40 NEXT n
```

En el programa anterior la cadena alfanumérica «MICROHOBBY» es visualizada 61 veces; se utiliza para formatear, el signo ortográfico «;».

La variable de control puede ser incluida en el grupo de sentencias que forman el bucle.

Ejemplo:

```
10 REM *************
       *           *
       * CUADRADOS *
       *           *
       *************
20 FOR c=100 TO 199
30 PRINT "Numero: ";c,
40 LET a=c↑2
50 PRINT "Cuadrado: ";a
60 NEXT c
```

la variable de control «C» que varía entre 1ØØ y 199 es elevada al cuadrado, asignando este valor a la variable «a»; ambas variables son visualizadas.

Los límites, al ser expresiones de tipo numérico, pueden estar constituidos por variables previamente asignadas.

Ejemplo:

```
10 REM **************
       *            *
       * PARAMETROS *
       *            *
       **************
20 INPUT "Limite inferior >>> ";inferior
30 INPUT "Limite superior >>> ";superior
40 IF superior<inferior THEN GO TO 20
50 FOR n=inferior TO superior
60 PRINT "Inferior = ";inferior;"  ";
70 PRINT "Superior = ";superior
80 NEXT n
```

En este programa, los parámetros o límites son introducidos con sentencias «INPUT»; la línea 4Ø verifica que el valor asignado a la variable «superior» es mayor que el de la variable «inferior», en caso contrario, han de introducirse, de nuevo, los límites.

Dentro de un bucle se puede modificar el valor de la variable de control.

Ejemplo:

```
10 REM ***************
       *             *
       * MODIFICAR 1 *
       *             *
       ***************
20 FOR x=10 TO 40
30 LET x=x-2
40 PRINT "Valor de x: ";x
50 NEXT x
```

Como se puede observar, al ejecutar el programa, la variable «X» no alcanza los valores asignados en los límites, ya que en la línea 3Ø se decrece en dos su valor, ¿por qué al visualizar «X» se decrece sólo en una unidad?, la respuesta es sencilla, ya que la instrucción «LET X = X — 2» decrece dos unidades, pero la sentencia «NEXT X» incrementa en uno su valor, luego —2 + 1 = 1.

Cuando se modifican los límites en el interior de un bucle, el intérprete BASIC hace caso omiso de los nuevos valores y lo ejecuta con los iniciales.



Ejemplo:

```
10 REM ***********
       *         *
       * MODIF 2 *
       *         *
       ***********
20 LET lim inf=1
30 LET lim sup=20
40 FOR f=lim inf TO lim sup
50 LET lim inf=30
60 LET lim sup=90
70 PRINT "Control: ";f,"Limites: ";lim inf;" y ";lim sup
80 NEXT f
```

A pesar de las líneas 5Ø y 6Ø que modifican los límites, la variable «f» asume los valores iniciales 1 a 2Ø.

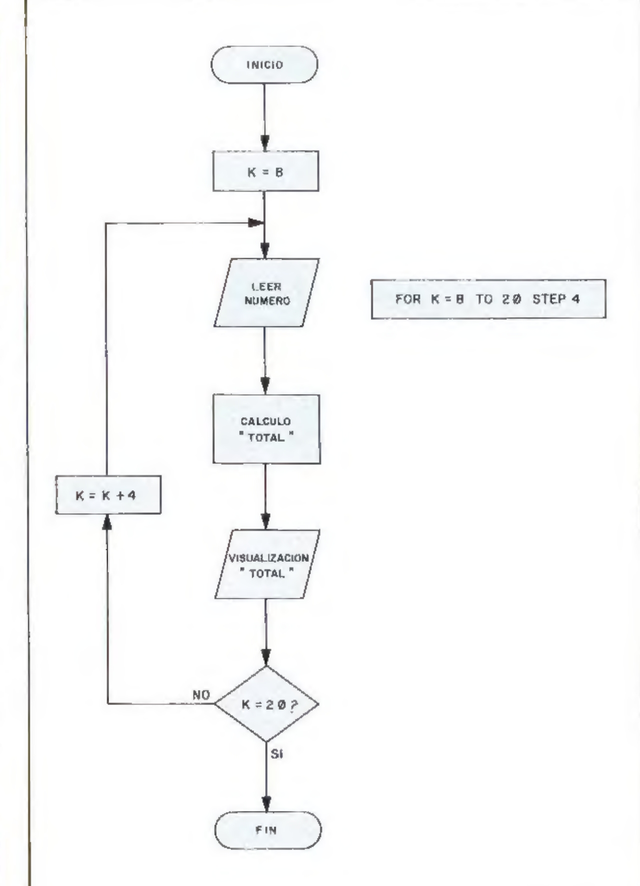
Dentro de un bucle «FOR/NEXT» se puede incluir una instrucción del tipo «IF ... THEN ...» que si se cumple la condición, se produzca una ruptura en su ejecución, aunque la variable de control no haya alcanzado su valor final.

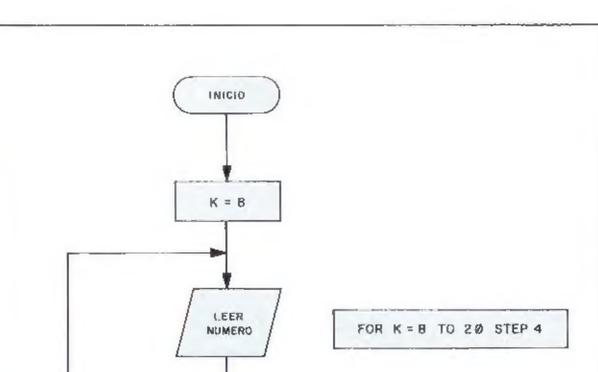
Este tipo de estructura de bucle es similar a la que, en otros lenguajes de más alto nivel, se denomina «DO WHILE/BREAK».

Ejemplo:

```
10 REM *********
       *       *
       * BREAK *
       *       *
       *********
20 FOR w=1 TO 15
30 INPUT "Radio >>> ";radio
40 IF radio=0 THEN GO TO 80
50 LET area=PI*radio↑2
60 PRINT "Radio: ";radio,"Area: ";area
70 NEXT w
80 PRINT '''"************** FIN **************"
```

si se introducen valores de radio distintos de «Ø», el bucle se ejecuta un número de veces que coincide con los parámetros especificados (15 veces); si por el contrario, se introduce el *código de ruptura*, que en este caso es igual a «Ø»; la ejecución pasa a la línea 1Ø sin haber concluido el bucle.





Desarrollo de un bucle FOR-NEXT.

STEP

**Acceso al teclado**




### Tipo de sentencia

Comando de programación.

### Definición

La palabra clave «STEP» se maneja conjuntamente con la sentencia «FOR ... TO...», formando parte de su argumento. Se utiliza para modificar el *paso* de un bucle, es decir, el incremento que se realiza a la variable de control cuando se ejecuta la sentencia «NEXT» correspondiente.

# BUCLES

Al analizar un programa, sucede con bastante frecuencia, que deba repetirse un cálculo o realizar una misma tarea con distintos datos. Sería una manera ilógica, en principio, editar las instrucciones de cálculo tantas veces como datos tengamos, ya que se utilizaría más memoria del ordenador; una forma algo más lógica sería utilizar un *bucle* (loop en inglés), que repitiera las mismas instrucciones tantas veces como quisieramos.

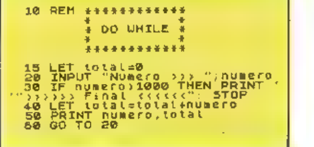
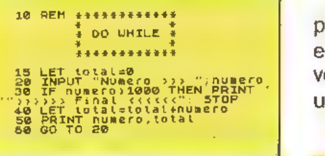
En programación, la terminología inglesa de los tipos de bucle más utilizada es:

— DO WHILE
— REPEAT UNTIL
— FOR ... NEXT

## DO WHILE

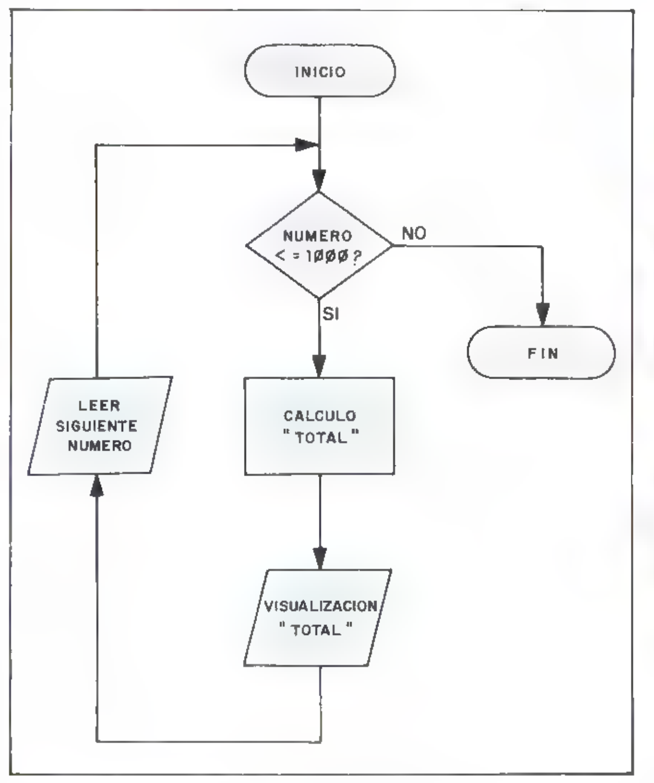
La estructura de bucle DO WHILE (Hacer mientras...) permite realizar una tarea varias veces, siempre y cuando se cumpla la condición impuesta.

Ejemplo:

```
10 REM **************
       *            *
       * DO WHILE   *
       *            *
       **************
15 LET total=0
20 INPUT "Numero >>> ";numero
30 IF numero>1000 THEN PRINT "
">>>>>> Final <<<<<<": STOP
40 LET total=total+numero
50 PRINT numero,total
60 GO TO 20
```

mientras se cumpla que la variable «número» sea igual o inferior a «1ϕϕϕ», se realizarán indefinidamente las tareas de cálculo y visualización de la variable «total».




Estructura «DO WHILE».

## REPEAT UNTIL

Con este tipo de bucle, (Repetir hasta..., es la traducción) el ordenador realiza una y otra vez la tarea hasta que se da una condición.

Ejemplo:

```
10 REM **************
       *            *
       * DO UNTIL   *
       *            *
       **************
15 LET total=0
20 INPUT "Numero >>> ";numero
30 LET total=total+numero
40 PRINT numero,total
50 IF total>1000 THEN PRINT "
">>>>>> FIN <<<<<<": STOP
60 GO TO 20
```

en este otro ejemplo, se repite el bucle hasta que se cumple la condición de que la variable «total» es superior a «1ϕϕϕ».

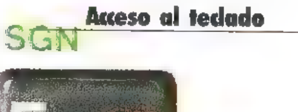
## Diferencias

La diferencia entre estos dos tipos de estructura estriba, en que en el primer caso (DO WHILE) la salida del bucle se encuentra antes de realizar la tarea, y en el segundo (REPEAT UNTIL) se encuentra al final, ¿Qué significado práctico tiene esto? Que con una estructura DO WHILE, si al entrar en el bucle no se cumple la condición prevista, se sale de él sin haber ejecutado ninguna tarea; sin embargo, con la estructura REPEAT UNTIL, al menos *una vez* se ejecutan las instrucciones contenidas en él.

FOR/NEXT

## Acceso al teclado

SGN




MODO K



IN KEY $




MODO K

OVER

## Tipo de sentencia

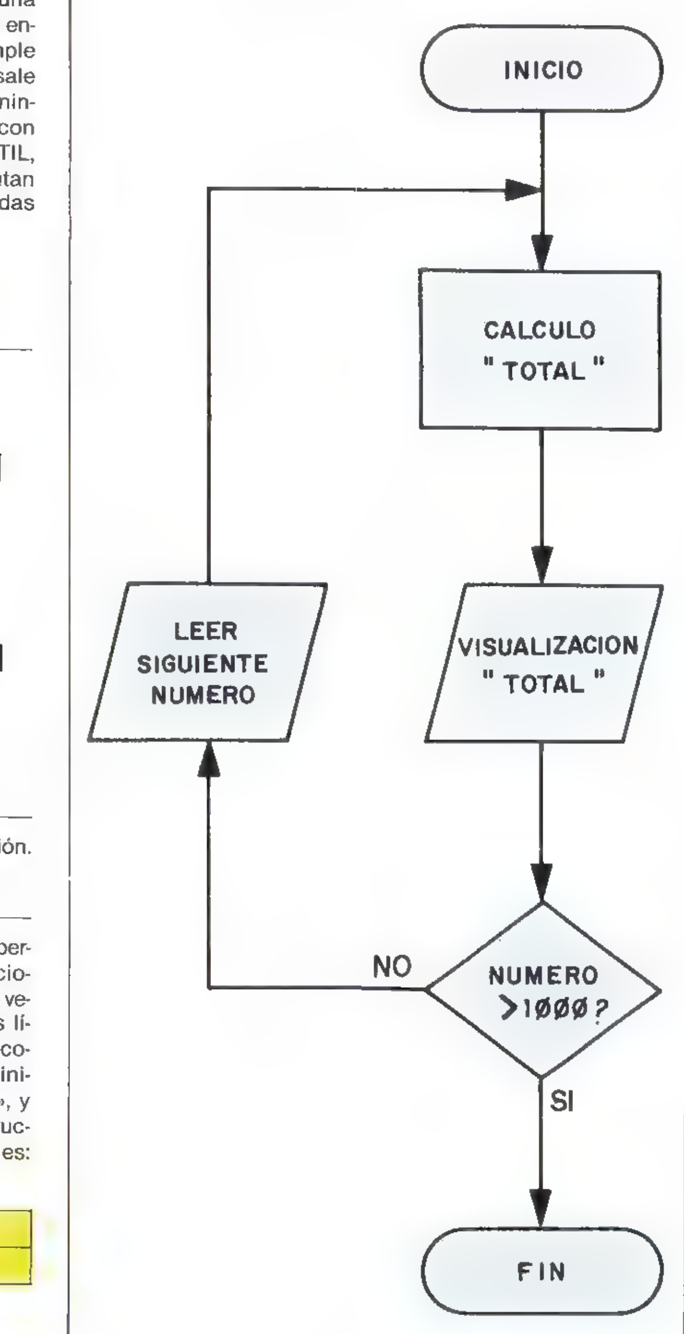
Comando de programación.

## Definición

Los bucles «FOR/NEXT» permiten realizar las instrucciones contenidas en él tantas veces como se indica en los límites del argumento. El comienzo del bucle está definido por la sentencia «FOR», y el final por «NEXT». La estructura de la sentencia «FOR» es:

| SENTENCIA | ARGUMENTO |
| --- | --- |
| FOR | var.= exp.TO exp. |

var. = variable.
exp. = expresión.
dentro del argumento se utili-




Estructura «REPEAT UNITL».

**Una guerra en las galaxias**

## CODENAME MAT






*Micromega/Ventamatic*
*48 K*
*Tipos de juego:*
*PVP: 1.700*






Este juego en tres dimensiones está basado en una pequeña historia que nos hace recordar un poco a las famosas epopeyas galácticas del cine. Los Myons, una raza que ha intentado dominar durante siglos el sistema solar, han lanzado un ataque contra nosotros. Para impedir que logren su objetivo tenemos una nave espacial revolucionaria que es además un centro táctico de mando.

Disponemos de un gran número de mando para controlar, que tenemos que manejar: controles de dirección y disparo; impulso y velocidad de las máquinas; túnel hiperlumínico; escudos defensivos; computadora de seguimiento; interruptor de seguimiento de naves enemigas; transmisor subespacial y los controles de Display.

Hay tres niveles de dificultad en el juego: el primero en el que podemos practicar, el segundo en el que se produce una invasión moderada de Myons, y el tercero con una invasión total. Además de estas posibilidades hay otras dos más. Una en la que podemos actuar como comandantes de una flota y otra en la que únicamente somos pilotos.

Cada área o sector puede tener un determinado número de enemigos, cada uno de los cuales tendrá una forma característica de actuación. Son los Cazas, los Cruceros y la Estrella Base.

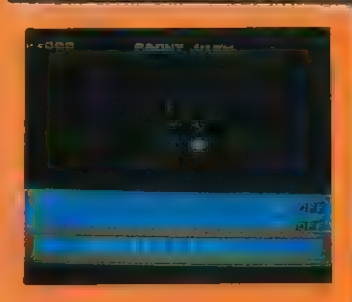
Nuestra misión defensiva debe llevarnos a intentar por todos los medios impedir la destrucción del sistema planetario que está bajo nuestro dominio. En la pantalla se nos va mostrando la información necesaria sobre el estado de éstos comunicándonos si ha sido destruido alguno de ellos.

El manual de instrucciones de este juego es realmente completo y nos explica con todo detalle todos y cada uno de los controles que tenemos que intentar manejar con el máximo de destreza posible, que son un total de veintiuno.

Dentro del juego es muy importante el dominio absoluto de todas las posibilidades que tenemos, ya que este será el único modo de conseguir cumplir con éxito nuestra misión, sobre todo en lo que se refiere al uso de los scaner de largo alcance o al computador de batalla.

Cuando uno juega con este programa llega a sentir la sensación de estar librando una batalla real en el espacio y se siente el protagonista de una película galáctica, los gráficos, la ambientación y el modo general en el que ha sido desarrollado el juego, contribuyen a hacérnoslo creer.

Es un juego muy bueno que ha aprovechado el camino empezado por el Time Gate y ha conseguido crear un programa de alto nivel, que resulta francamente muy entretenido.



## LOS MAS VENDIDOS

| | | |
|---|---|---|
| 1. | Decathlon | Ocean |
| 2. | Match Point | Psion |
| 3. | Knight Lore | Ultimate |
| 4. | SabreWulf | Ultimate |
| 5. | Underwurlde | Ultimate |
| 6. | Full Throttle | Micromega |
| 7. | Beach Head | Gold U.S. |
| 8. | Fighter Pilot | Digital |
| 9. | Combat Linx | Durell |
| 10. | TLL | Vortex |

**LISTA ELABORADA CON LA COLABORACION DE:**

Key Informática
Microbyte
Geco Informática
Tronik

# TEST

**Premiado con 15.000 ptas.**

*Benjamín JIMENEZ*

**Spectrum 48 K**

**Con este programa podrá elaborar todo tipo de test, del modo más simple y ágil. Siguiendo las instrucciones que le ofrecemos a continuación, conseguirá, sin esfuerzo, el más complejo sondeo que se proponga.**

El primer paso a seguir será el de seleccionar el menú que desee, con las siguientes opciones:

1. ELABORAR TEST
   Pide: Nombre del test para posterior identificación.
   Número de preguntas de que consta el test (1 a 20).
   Número de respuestas posibles (1 a 4) por pregunta.
   Número de la respuesta correcta de esa pregunta.
2. EJECUTAR EL TEST
   Pasa a ejecutar el test seleccionado (opción 5), dejando entre pregunta y pregunta, un tiempo concreto (opción 4) y diciendo si la respuesta es correcta o errónea.
3. SALVAR EL TEST
   Salva el programa y los datos correspondientes a los test creados con el nombre que se desee.
4. CAMBIO DE VELOCIDAD
   Altera el tiempo que deja el Spectrum para responder a cada una de las preguntas del test.
5. SELECCIONAR TEST
   Aparece un menú con los nombres (1 a 4) de los test que hay creados en este momento, para que el usuario elija cuál de ellos desea ejecutar o modificar.
6. MODIFICAR EL TEST
   Permite modificar cualquier pregunta o respuesta del test seleccionado (opción 5).
7. TERMINAR
   Stop.

Al terminar la ejecución de cada test (opción 2), el programa nos da un resumen de las respuestas acertadas y las preguntas de que constaba el test.

El programa está salvado con autoejecución en la línea 30, para respetar el contenido de las matrices creadas en la línea 6.000. Si desea limpiar este contenido y comenzar el programa, teclear RUM.



J.R. BALLESTEROS

```
1 LET F=0: LET PREG=0
10 BORDER 1: PAPER 1: INK 7
12 POKE 23658,8: POKE 23609,100
15 IF f=0 THEN GO SUB 6000
20 REM ******* MENU PRINCIPAL ****
30 CLS : PRINT INVERSE 1;"        T E S T        "
40 PRINT AT 3,3; INVERSE 1;" 1 - ELABORAR TEST    "
41 PRINT AT 5,3; INVERSE 1;" 2 - EJECUTAR EL TEST  "
42 PRINT AT 7,3; INVERSE 1;" 3 - SALVAR EL TEST   "
43 PRINT AT 9,3; INVERSE 1;" 4 - CAMBIAR VELOCIDAD  "
45 PRINT AT 11,3; INVERSE 1;"5 - SELECCIONAR TEST  "
47 PRINT AT 13,3; INVERSE 1;"6 - MODIFICAR LOS TEST "
49 PRINT AT 15,3; INVERSE 1;"7 - TERMINAR         "
50 PRINT AT 21,0;"SELECCIONA OPCION": PAUSE 0
55 IF INKEY$="1" THEN GO TO 100
56 IF INKEY$="3" THEN GO TO 2000
57 IF INKEY$="2" THEN GO TO 365
58 IF INKEY$="4" THEN GO TO 1000
59 IF INKEY$="5" THEN GO SUB 3000: GO TO 30
60 IF INKEY$="6" THEN GO TO 4000
65 IF INKEY$="7" THEN STOP
80 GO TO 55
99 REM *********** ELABORAR TEST **
100 CLS
101 GO SUB 3000: CLS
102 LET M$="     ELABORAR TEST: ": LET N$="PREG: ": LET O$="RESP: "
103 GO SUB 8000
105 INPUT "NOMBRE DEL TEST: ";X$: LET Z$(TEST)=X$
106 GO SUB 8000
109 INPUT "Dime el numero de preguntas de  que consta el test ";x$
110 GO SUB 520: IF c=1 THEN GO TO 100
120 IF VAL x$>50 OR VAL x$<1 THEN GO TO 100
130 LET preg=VAL x$
135 GO SUB 8000
140 INPUT "Dime el numero de respuestas    posibles ";x$
150 GO SUB 520: IF c=1 THEN GO TO 140
160 IF VAL x$>4 OR VAL x$<1 THEN GO TO 140
170 LET res=VAL x$
180 IF TEST=1 THEN DIM A$(preg,60): DIM B$(preg,res,25)
181 IF TEST=2 THEN DIM E$(preg,60): DIM F$(preg,res,25)
182 IF TEST=3 THEN DIM G$(preg,60): DIM H$(preg,res,25)
183 IF TEST=4 THEN DIM I$(preg,60): DIM J$(preg,res,25)
190 FOR i=1 TO preg
200 GO SUB 8000: LET x$=" "
210 INPUT "Dame el texto de pregunta ";(i);"      ";x$
220 IF LEN x$>60 THEN LET x$=x$( TO 60)
230 PRINT AT 4,0; INVERSE 1;" ";i;" - ";x$;" "
240 IF TEST=1 THEN LET A$(i)=x$: GO TO 250
241 IF TEST=2 THEN LET E$(i)=x$: GO TO 250
242 IF TEST=3 THEN LET G$(i)=x$: GO TO 250
243 IF TEST=4 THEN LET I$(i)=x$
250 FOR j=1 TO res
260 INPUT "Dame texto respuesta ";(j);" ";"de la Preg ";(i);" ";x$
270 PRINT ''TAB 5;j;" - ";x$
280 IF LEN x$>24 THEN LET x$=x$( TO 24)
290 IF TEST=1 THEN LET B$(i,j)=x$: GO TO 300
291 IF TEST=2 THEN LET F$(i,j)=x$: GO TO 300
292 IF TEST=3 THEN LET H$(i,j)=x$: GO TO 300
293 IF TEST=4 THEN LET J$(i,j)=x$
300 NEXT j
310 INPUT "Dime el numero de la respuesta  correcta ";x$
320 GO SUB 520: IF c=1 THEN GO TO 310
330 IF VAL x$>res OR VAL x$<1 THEN GO TO 310
340 IF TEST=1 THEN LET B$(i,VAL x$)(25)="*": GO TO 360
341 IF TEST=2 THEN LET F$(i,VAL x$)(25)="*": GO TO 360
342 IF TEST=3 THEN LET H$(i,VAL x$)(25)="*": GO TO 360
343 IF TEST=4 THEN LET J$(i,VAL x$)(25)="*"
360 NEXT i
361 GO SUB 5000: LET Z(TEST)=PREG: LET Y(TEST)=RES
362 GO TO 30
365 REM ******** PROBAR EL TEST ++
366 LET M$="        T E S T : "
370 LET BIEN=0: FOR i=1 TO preg
375 CLS : GO SUB 8007
380 IF TEST=1 THEN PRINT AT 2,2-LEN STR$ i; INVERSE 1;i;"  ";A$(i): GO TO 390
381 IF TEST=2 THEN PRINT AT 2,2-LEN STR$ i; INVERSE 1;i;"  ";E$(i): GO TO 390
382 IF TEST=3 THEN PRINT AT 2,2-LEN STR$ i; INVERSE 1;i;"  ";G$(i): GO TO 390
383 IF TEST=4 THEN PRINT AT 2,2-LEN STR$ i; INVERSE 1;i;"  ";I$(i)
390 PRINT : FOR j=1 TO res
400 IF TEST=1 THEN PRINT 'TAB 3;j;" - ";B$(i,j)( TO 24): GO TO 410
401 IF TEST=2 THEN PRINT 'TAB 3;j;" - ";F$(i,j)( TO 24): GO TO 410
402 IF TEST=3 THEN PRINT 'TAB 3;j;" - ";H$(i,j)( TO 24): GO TO 410
403 IF TEST=4 THEN PRINT 'TAB 3;j;" - ";J$(i,j)( TO 24)
410 NEXT j
420 PRINT AT 21,0;"Cual es la correcta ? "
425 FOR K=1 TO T
426 IF INKEY$<>"" THEN GO TO 439
428 NEXT k: BEEP .15,14: BEEP .15,8: BEEP .5,16
429 GO TO 475
430 IF INKEY$="" THEN GO TO 0430
439 IF INKEY$="M" THEN GO TO 30
440 LET x$=INKEY$: GO SUB 520: IF c=1 THEN GO TO 440
450 IF VAL x$>res OR VAL x$<1 THEN GO TO 0430
460 IF TEST<>1 THEN GO TO 463
461 IF B$(i,VAL x$)(25 TO 25)="*" THEN GO SUB 490: GO TO 475
462 GO SUB 500
463 IF TEST<>2 THEN GO TO 466
464 IF F$(i,VAL x$)(25 TO 25)="*" THEN GO SUB 490: GO TO 475
465 GO SUB 500
466 IF TEST<>3 THEN GO TO 469
467 IF H$(i,VAL x$)(25 TO 25)="*" THEN GO SUB 490: GO TO 475
468 GO SUB 500
469 IF TEST<>4 THEN GO TO 472
470 IF J$(i,VAL x$)(25 TO 25)="*" THEN GO SUB 490: GO TO 475
471 GO SUB 500
475 NEXT I
478 CLS : PRINT AT 10,0;BIEN;" RESPUESTAS ACERTADAS DE ";PREG: PRINT AT 21,0;"PULSA UNA TECLA PARA CONTINUAR": PAUSE 0
480 GO TO 30
490 CLS : PRINT AT 9,10; INVERSE 1;"    BIEN !!!    ": PAUSE 150: LET BIEN=BIEN+1: CLS : RETURN
500 CLS : PRINT AT 9,10; INVERSE 1;"    MAL  **** ": PAUSE 150: CLS : RETURN
510 STOP
520 IF x$="" THEN LET c=1: RETURN
530 FOR h=1 TO LEN x$
540 IF CODE x$(h)<48 OR CODE x$(h)>57 THEN LET c=1: RETURN
550 NEXT h
560 LET c=0
570 RETURN
1000 CLS : PRINT AT 10,5;"VELOCIDAD ACTUAL: ";T
1010 INPUT "NUEVA VELOCIDAD: ";X$
1012 GO SUB 520: IF C<>0 THEN GO TO 1000
1015 LET T=VAL X$
1020 PRINT AT 12,5;"NUEVA VELOCIDAD : ";T
1025 PAUSE 150
1030 GO TO 30
2000 CLS : INPUT "NOMBRE: ";X$
2010 IF LEN X$>10 THEN LET X$=X$( TO 10)
2020 SAVE X$ LINE 10
2030 CLS : PRINT AT 10,5;"REBOBINA PARA VERIFICAR": VERIFY ""
2040 GO TO 30
3000 REM ****** SELECCIONAR TEST **
3010 CLS
3012 PRINT INVERSE 1;"        SELECCION DE TEST        "
3015 FOR I=1 TO 4: PRINT AT I*2+2,6;I;".- ";Z$(I): NEXT I
3020 PRINT AT 21,0;"NO. DE TEST A PROBAR O CAMBIAR: ": PAUSE 0
3030 LET X$=INKEY$
3031 GO SUB 520: IF C<>0 THEN GO TO 3020
3032 IF VAL X$<1 OR VAL X$>4 THEN GO TO 3020
3033 CLS
3040 LET TEST=VAL X$
3041 LET X=VAL X$
3050 LET preg=z(test): LET res=y(test)
3900 RETURN
4000 REM modificar test **********
4010 GO SUB 3000
4020 LET m$="   MODIFICAR TEST: "
4030 GO SUB 8000
4035 IF PREG=0 THEN PRINT FLASH 1;AT 12,0;"  ANTES DEBES ELABORAR EL TEST  ": PAUSE 250: GO TO 30
4040 INPUT "NO. DE PREGUNTA ? (M=MENU) ";X$: IF X$="M" THEN GO TO 30
4045 GO SUB 520: IF C<>0 THEN GO TO 4040
4050 IF VAL X$<1 OR VAL X$>PREG THEN GO TO 4040
4059 LET P=VAL X$
4060 IF TEST=1 THEN PRINT AT 4,2-LEN STR$ P;P;"  "; INVERSE 1;A$(P): GO TO 4070
4061 IF TEST=2 THEN PRINT AT 4,2-LEN STR$ P;P;"  "; INVERSE 1;E$(P): GO TO 4070
4062 IF TEST=3 THEN PRINT AT 4,2-LEN STR$ P;P;"  "; INVERSE 1;G$(P): GO TO 4070
4063 IF TEST=4 THEN PRINT AT 4,2-LEN STR$ P;P;"  "; INVERSE 1;I$(P): GO TO 4070
4070 INPUT "CORRECTO ? ";X$
4071 IF X$(1)="S" THEN GO TO 4100
4072 IF X$(1)="N" THEN GO TO 4080
4073 GO TO 4070
4080 INPUT "NUEVO TEXTO:";X$
4085 IF TEST=1 THEN LET A$(P)=X$
4086 IF TEST=2 THEN LET E$(P)=X$
4087 IF TEST=3 THEN LET G$(P)=X$
4088 IF TEST=4 THEN LET I$(P)=X$
4090 GO TO 4060
4100 FOR I=1 TO RES: PRINT ';TAB 4;I;" - ";
4105 IF TEST=1 THEN PRINT B$(P,I)
4106 IF TEST=2 THEN PRINT F$(P,I)
4107 IF TEST=3 THEN PRINT H$(P,I)
4108 IF TEST=4 THEN PRINT J$(P,I)
4110 NEXT I
4115 INPUT "CORRECTO ?";X$
4116 IF X$(1)="S" THEN GO TO 4200
4117 IF X$(1)="N" THEN GO TO 412■
4118 GO TO 4115
4120 INPUT "PREGUNTA ? (0 = FIN) ";X$: GO SUB 520: IF C<>0 THEN GO TO 4120
4125 IF VAL X$>RES THEN GO TO 4120
4130 IF X$="0" THEN GO TO 4020
4135 LET R=VAL X$
4140 INPUT "NUEVO TEXTO: ";X$
4150 IF TEST=1 THEN LET B$(P,R)=X$
4151 IF TEST=2 THEN LET F$(P,R)=X$
4152 IF TEST=3 THEN LET H$(P,R)=X$
4153 IF TEST=4 THEN LET J$(P,R)=X$
4154 GO SUB 8000: PRINT AT 5,0;" ": GO TO 4100
4200 FOR I=1 TO RES
4205 IF TEST<>1 THEN GO TO 4210
4206 IF B$(P,I)(25)="*" THEN LET B=I: GO TO 4250
4210 IF TEST<>2 THEN GO TO 4220
4211 IF F$(P,I)(25)="*" THEN LET B=I: GO TO 4250
4220 IF TEST<>3 THEN GO TO 4230
4221 IF H$(P,I)(25)="*" THEN LET B=I: GO TO 4250
4230 IF TEST<>4 THEN GO TO 4240
4231 IF J$(P,I)(25)="*" THEN LET B=I: GO TO 4250
4240 NEXT I
4250 INPUT "RESPUETA CORRECTA LA NO. ";(B);"         CORRECTO ? ";X$
4255 IF X$(1)="S" THEN GO TO 4300
4257 IF X$(1)="N" THEN GO TO 4260
4258 GO TO 4250
4260 INPUT "NO. RESPUESTA CORRECTA ? ";X$: GO SUB 520: IF C<>0 THEN GO TO 4260
4265 IF VAL X$<1 OR VAL X$>RES THEN GO TO 4260
4270 IF TEST=1 THEN LET B$(P,B)(25)=" ": LET B$(P,VAL X$)(25)="*"
4271 IF TEST=2 THEN LET F$(P,B)(25)=" ": LET F$(P,VAL X$)(25)="*"
4272 IF TEST=3 THEN LET H$(P,B)(25)=" ": LET H$(P,VAL X$)(25)="*"
4273 IF TEST=4 THEN LET J$(P,B)(25)=" ": LET J$(P,VAL X$)(25)="*"
4280 LET B=VAL X$: GO TO 4250
4300 GO TO 4020
5000 REM ****** CARGAR MATRICES **
5100 RETURN
6000 REM ****** inicializar ******
6010 DIM Z$(4,10): REM NOMBRE TEST
6015 DIM Z(4): REM PREGUNTAS TEST
6020 DIM Y(4): REM RESPUESTAS TEST
6030 LET F=1
6050 LET T=400: BORDER 1: PAPER 1: INK 7: CLS
6055 POKE 23609,200
6060 RETURN
8000 CLS
8005 PRINT AT 2,20; INVERSE 1;N$;PREG;"  "
8006 PRINT AT 3,20; INVERSE 1;O$;RES;"  "
8007 PRINT AT 0,0; INVERSE 1;M$;"  ";Z$(TEST)
8010 RETURN
9999 PRINT 65561-USR 7962
```

Una vez más y siguiendo con nuestro sorteo mensual de un MICRODRIVE y su INTERFACE 1 correspondiente, hemos procedido a la elección del galardonado en el mes de diciembre, siendo el afortunado JOSE CARLOS PARRA REAÑO, que vive en la calle SANTIAGO RUSIÑOL, 2 de ARANJUEZ (MADRID).

Como saben, este sorteo se realiza entre todos nuestros lectores que quieran enviarnos un programa, de fabricación propia, para publicar en la sección que lleva el mismo nombre. Anímense y, en caso de ser publicado, recibirán, además, 15.000 pesetas.

# LOS INVASORES

*Antonio y Pilar FERNANDEZ CANTON*

**Spectrum 48 K**

**Premiado con 15.000 ptas.**

**Estamos ante otro juego espacial en el que nos situamos en una nave para cumplir una misión crucial para el futuro de nuestro planeta.**

Debemos defender, lo mejor posible, nuestra zona del ataque de constantes patrullas de ovnis que intentan buscar nuestros puntos vulnerables y atacar. Si dejamos pasar diez de éstos, nuestro planeta estallará irremisiblemente. Por cada veinticinco ovnis que logremos eliminar, nuestra nave accederá a otro sector e iremos superando el trance.

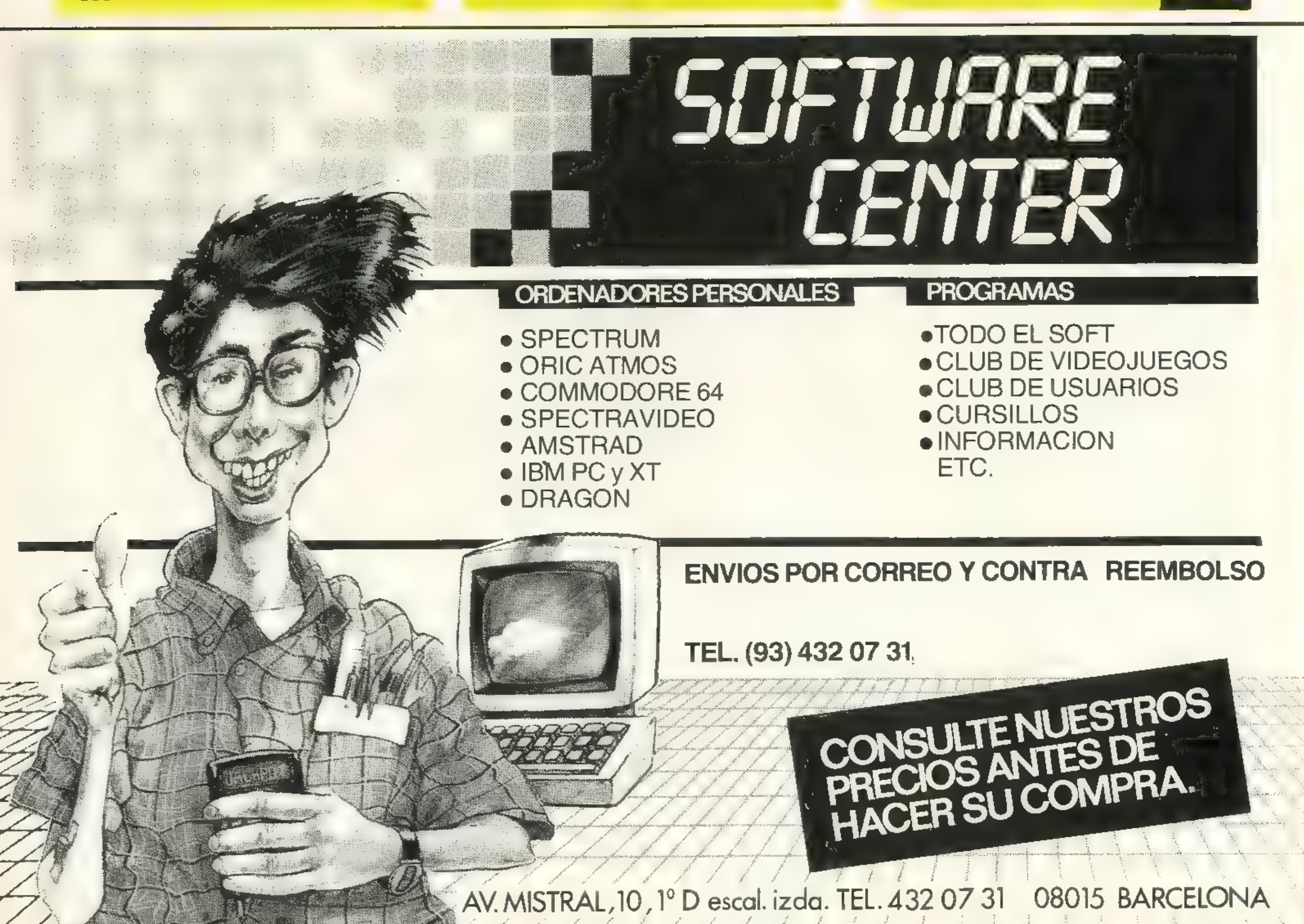
Pero ¡cuidado! eso no es tan fácil, ya que el movimiento de los objetos volantes no identificados, es imprevisible.

NOTAS GRAFICAS

| A | B | C | D | E | F |
|---|---|---|---|---|---|
| ▲ | ▬ | △ | ✱ | ✱ | ✱ |

```
 130 FOR f=USR "a" TO USR "a"+47
 140 READ a: POKE f,a: NEXT f
 150 DATA 0,224,112,126,249,255,
255,0,0,0,0,0,252,255,252,0
 155 BORDER 1: PAPER 0: INK 7: B
RIGHT 1: CLS
 160 DATA 0,24,36,36,102,129,129
,126
 163 DATA 0,112,16,108,31,103,31
,116
 165 DATA 74,74,169,55,223,47,20
2,20
 167 DATA 84,169,174,232,253,234
,100,146
 170 FOR y=1 TO 100: PLOT RND*25
5,RND*166: NEXT y
 180 INPUT "Nivel de dificultad
1,9?";d: IF d>9 OR d<1 THEN GO T
O 180
 185 LET d=10-d
 190 LET d=d/100
 200 LET x=INT (RND*21)+1: LET ■
=0: LET t=0: LET px=31.3: LET py
=INT (RND*21)+1
 210 LET g=0
 225 GO SUB 500
 230 PRINT INK 6;AT x,0;"▲"
 233 BEEP .5,-20
 235 IF INKEY$="" THEN GO TO 235
 237 PRINT OVER 1;AT py,px;"△"
 240 PRINT OVER 1;AT py,px,"△"
 250 LET px=px-1: LET py=py-(2 A
ND RND>.5 AND py>=2)+(1 AND py<2
1)
 260 PRINT OVER 1;AT py,px,"△"
 270 IF m=10 THEN GO SUB 650
 280 IF px<=2.5 THEN LET m=m+1:
PRINT AT py,px;" ": GO SUB 500:
LET px=31.3: BEEP .1,20: LET py=
INT (RND*21)+1: GO TO 260
 300 IF INKEY$="" THEN BEEP d,50
-px: GO TO 240
 310 IF INKEY$="q" AND x>1 THEN
PRINT AT x,0;" ": LET x=x-1: GO
 TO 330
 320 IF INKEY$="s" AND x<20 THEN
 PRINT AT x,0;" ": LET x=x+1
 330 PRINT INK 6;AT x,0;"▲"
 350 IF INKEY$<>"f" THEN BEEP d,
50-px: GO TO 240
 360 PLOT OVER 1;16,8*(21-x)+1:
DRAW OVER 1;239,0
 370 BEEP .2,10
 380 PLOT OVER 1,16,8*(21-x)+1:
DRAW OVER 1,239,0
 385 IF x<>py THEN GO TO 240
 387 PRINT OVER 1;AT py,px;"△"
 390 BORDER 2: PRINT OVER 1;AT p
y,px;"✱": LET t=t+1: BEEP .5,-3
5: GO SUB 500: PRINT OVER 1;AT p
y,px;"✱": LET py=INT (RND*22):
LET px=31.3: LET g=0: BORDER 1
 425 IF INT (t/25)=t/25 THEN GO
TO 600
 450 GO TO 250
 500 PRINT PAPER 6; INK 2;AT 0,0
;"Escapados=";m;"Derribos=";t;"T
otal=";m+t: RETURN
 600 FOR r=0 TO 29
 605 IF r>0 THEN PRINT AT x,r-1;
" "
 610 PRINT AT x,r; INK 3;;"✱"; I
NK 6;"▲": BEEP .15125-(r/194),r
/1.5
 620 NEXT r
 625 IF t=150 THEN BEEP 1,13: PR
INT AT 10,7; FLASH 1; INK 6; PAP
ER 2;"  G A N A D O R  ": PRINT
#1;"Pulsa una tecla": PAUSE 0: R
UN
 630 PRINT AT x,29;"  ": LET ■=
INT (RND*21)+1: GO TO 230
 650 BEEP .5,35: FOR f=1 TO 100
 660 LET e=22560+INT (RND*641)
 670 POKE e,INT (RND*256)
 675 BEEP .01,f/2
 677 OUT 254,RND*255
 680 NEXT f
 690 PRINT AT 10,5; FLASH 1; INK
 6; PAPER 2;"   P E R D I S T E
 "
 700 PRINT #1;" Pulsa una tecla
para empezar": PAUSE 0: RUN
1000 PAPER 0: INK 6: BORDER 6: C
LS
1010 PAPER 0: INK 6: BORDER 6: C
LS
1030 RUN
```


J.R. BALLESTEROS




# GEOMETRA

P. GARCIA **Spectrum 16 K**

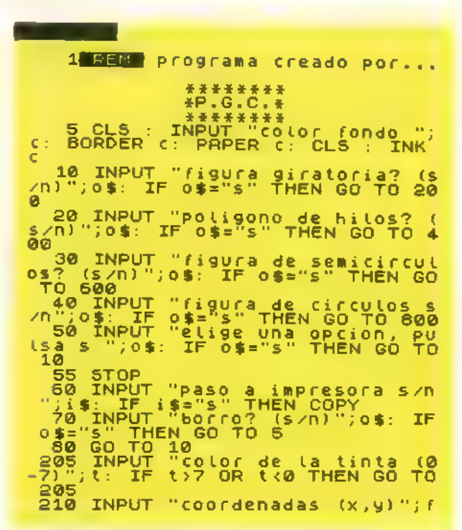
**Si quiere usar su Spectrum para resolver funciones trigonométricas este es el programa adecuado para ello.**

Se basa en las funciones seno y coseno, pudiéndose hacer por medio de este programa, figuras giratorias, polígonos de hilos, figuras de semicírculos y de círculos.

Para cargar el programa, sólo tiene que teclear LOAD " " o bien LOAD "pablo". Una vez establecido esto, tendrá que indicar al ordenador el color de fondo, la tinta y una serie de datos que le irá pidiendo.


A. PEREDA

**Premiado con 15.000 Ptas.**



```
   1 REM programa creado por...
           *********
           *P.G.C.*
           *********
   5 CLS : INPUT "color fondo ";
c: BORDER c: PAPER c: CLS : INK
c
  10 INPUT "figura giratoria? (s
/n)";o$: IF o$="s" THEN GO TO 20
0
  20 INPUT "poligono de hilos? (
s/n)";o$: IF o$="s" THEN GO TO 4
00
  30 INPUT "figura de semicircul
os? (s/n)";o$: IF o$="s" THEN GO
 TO 600
  40 INPUT "figura de circulos s
/n";o$: IF o$="s" THEN GO TO 800
  50 INPUT "elige una opcion, pu
lsa s ";o$: IF o$="s" THEN GO TO
 10
  55 STOP
  60 INPUT "paso a impresora s/n
";i$: IF i$="s" THEN COPY
  70 INPUT "borro? (s/n)";o$: IF
 o$="s" THEN GO TO 5
  80 GO TO 10
 205 INPUT "color de la tinta (0
-7)";t: IF t>7 OR t<0 THEN GO TO
 205
 210 INPUT "coordenadas (x,y)";f
;"  ,  ";g
 212 INPUT "radio ";r: IF f-r<0
OR f+r>255 OR g-r<0 OR g+r>175 T
HEN GO TO 210
 215 INPUT "n. lados ";n: IF n<3
 OR INT n<>n THEN GO TO 215
 230 INPUT "densidad ";d
 240 FOR a=0 TO 2*PI STEP (2*PI)
/d
 243 LET ax=f+COS a*r: LET ay=g+
SIN a*r
 246 PLOT ax,ay
 250 FOR b=1 TO n
 255 LET x=f+COS (a+b*2*PI/n)*r:
LET y=g+SIN (a+b*2*PI/n)*r
 260 DRAW INK t;x-ax,y-ay
 264 LET ax=x: LET ay=y
 270 NEXT b
 280 NEXT a
 290 GO TO 60
 405 INPUT "color de la tinta 0-
7";t: IF t>7 OR t<0 THEN GO TO 4
05
 410 INPUT "coordenadas (x-y) ";
f;"  ,  ";g
 420 INPUT "radio ";r
 430 IF f-r<0 OR f+r>255 OR g-r<
0 OR g+r>175 THEN GO TO 410
 440 INPUT "no. lados ";n: IF n<
3 OR INT n<>n THEN GO TO 440
 450 FOR a=0 TO 2*PI STEP 2*PI/r
 460 LET x=INT (f+COS a*r): LET
y=INT (g+SIN a*r)
 470 FOR b=1 TO n
 480 LET k=INT (f+COS (b*2*PI/n)
*r): LET l=INT (g+SIN (b*2*PI/n)
*r)
 490 PLOT x,y: DRAW INK t;k-x,l-
y
 500 NEXT b
 510 NEXT a
 520 GO TO 60
 605 INPUT "color tinta (0-7)";t
i: IF ti<0 OR ti>7 THEN GO TO 60
5
 610 INPUT "coordenadas (x,y)";f
;"  ,  ";g
 620 INPUT "radio  ";r
 630 IF f-r<0 OR f+r>255 OR g-r<
0 OR g+r>175 THEN GO TO 610
 640 INPUT "densidad ";d
 650 INPUT "angulo entre puntos
 (0-PI)";a: IF a>PI OR a<0 THEN G
O TO 650
 660 INPUT "angulo del trazado (
0-PI)";t: IF t>PI OR t<0 THEN GO
 TO 660
 670 FOR b=0 TO 2*PI STEP 2*PI/d
 680 LET x1=f+COS b*r: LET y1=g+
SIN b*r: LET x2=f+COS (b+a)*r: L
ET y2=g+SIN (b+a)*r
 690 PLOT x1,y1: DRAW INK ti;x2-
x1,y2-y1,-t
 700 NEXT b
 710 GO TO 60
 805 INPUT "color tinta (0-7)";t
: IF t<0 OR t>7 THEN GO TO 805
 810 INPUT "coordenadas (x,y)";f
;"  ,  ";g
 820 INPUT "distancia al centro
";d
 830 INPUT "radio ";r
 840 IF f-d-r<0 OR f+d+r>255 OR
g-d-r<0 OR g+d+r>175 THEN GO TO
810
 850 INPUT "densidad ";i
 860 FOR a=0 TO 2*PI STEP 2*PI/i
 870 LET x=f+d*COS a: LET y=g+d*
SIN a
 880 CIRCLE INK t;x,y,r
 890 NEXT a
 900 GO TO 60
```

# SOBRE LAS "NOTAS GRAFICAS"

**Aprovechando las numerosas llamadas de nuestros lectores preguntándonos el modo en el que se introducen los gráficos en todos los programas que publicamos en la revista, vamos a explicarlo de una forma general,**

1. La notas gráficas figuran en un recuadro de color que se encuentra situado justo al lado del programa.

2. Cada uno de los dibujos de este recuadro tiene una letra correspondiente a la cual pertenecen éstos.

3. Cuando encontremos en el listado del programa dicho dibujo, anotaremos a qué letra pertenece.

4. Procederemos a escribir la línea de forma normal, y una vez hayamos abierto las comillas, colocaremos el cursor en modo gráfico. Para conseguirlo, habrá que pulsar de forma simultánea las teclas CAPS SHIFT y 9. Una vez lo hayamos hecho, aparecerá el cursor en modo «G». Será entonces el momento de introducir la letra a la que correspondía dicho dibujo, tras lo cual, volveremos a pulsar nuevamente las teclas CAPS SHIFT y 9 para devolver el cursor al modo «L». Una vez hecho esto, cerraremos las comillas y continuaremos de forma hasta encontrar otro carácter gráfico.

5. Al introducir la línea con el gráfico correspondiente, no notaremos ningún cambio. No aparecerá en nuestro listado el dibujo que aparece en la revista, ya que para que esto ocurra, será necesario que antes se haya ejecutado el programa, momento en el cual lo que en principio eran letras se convertirán en dibujos. Si sacáramos el listado una vez que se ha ejecutado el programa, éste entonces, sí sería exactamente igual que el que vemos en la revista.

Lo máximo en protección de programas

# ASI ES EL SISTEMA «TURBO»

Gabriel NIETO

**Decir Pablo, Santiago y Emilio, para muchos puede suponer simplemente decir tres nombres más. Pero sin embargo, algunos de los usuarios de Spectrum los relacionarán rápidamente con un programa: el DECATHLON.**

Hace ya algunos meses, los tres juntos decidieron empezar una aventura que se presentaba excitante: romper una muralla famosa, la que separaba a muchos usuarios del acceso al interior de un codiciado programa. Sobre ella se enarbolaba una bandera con un nombre, Turbo, la diferencia entre poder y no poder.

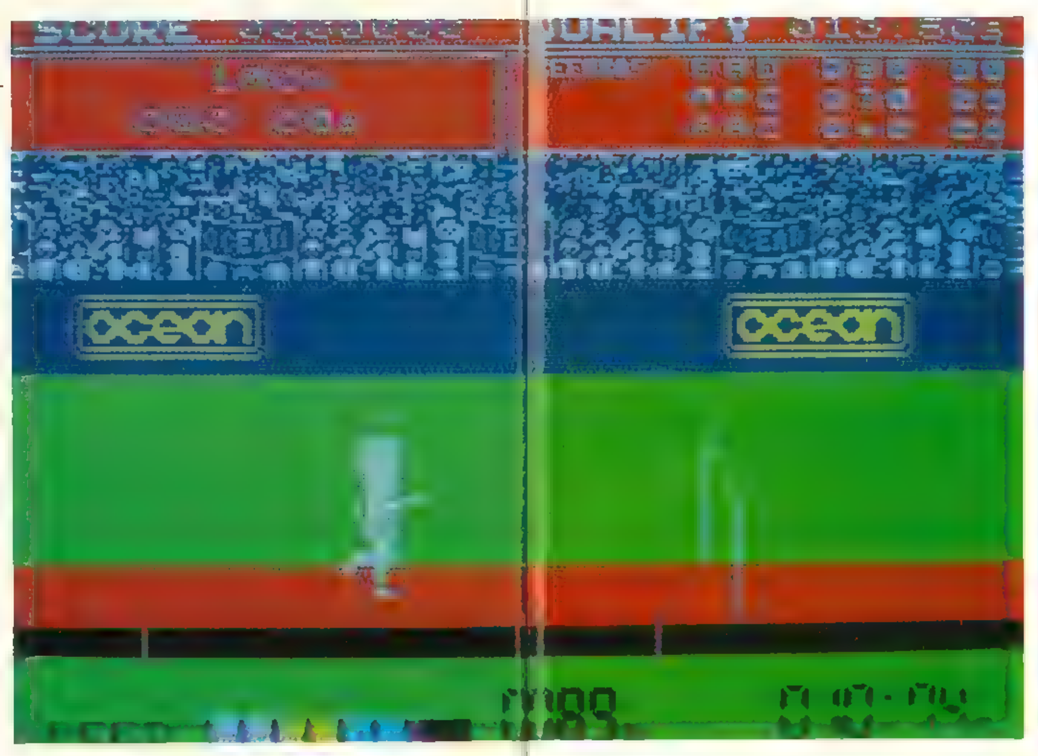
El Software es, para el usuario, lo mismo que el pan para el hambriento. Constituye el principal alimento de nuestro ordenador. Lo que ocurre, al igual que con el cesto de la compra, es que los precios se disparan, y cada vez se hace más difícil el poder alimentar a nuestro ordenador. Un ordenador que, por otra parte, necesita devorar programas para conseguir la energía necesaria, que nos da la dimensión auténtica de su potencia.

Un programa normal de juegos, tiene un precio medio que oscila en torno a las 2.000 pts., un precio muy alto si lo comparamos, por ejemplo, con el software inglés, y mucho más alto aún si tenemos en cuenta el bolsillo del usuario español, que en una gran mayoría suelen ser jóvenes con escasos recursos económicos.

La realidad, sin embargo, es que el Software en España se ha convertido en un negocio de dimensiones sorprendentes, los intereses son muchos y la necesidad de proteger unos sustanciosos beneficios comerciales se hace cada vez mayor. Por eso, las grandes compañías insisten en reforzar la protección de sus programas con el fin de convertirlos en auténticas fortalezas inexpugnables. La relación entre el usuario y su ordenador se ve quebrantada por este motivo. Ya no nos responde cuando le preguntamos. La palabra LIST que tantas veces utilizamos con la morbosa, pero sana curiosidad, de querer conocer qué se esconde tras aquel curioso programa, para ver de cerca el mecanismo que produce esos efectos en nuestra pantalla de TV, se ha convertido en una pieza inútil. Ni siquiera nuestra fiel BREAK nos responde, los fabricantes le han hecho enmudecer, pero no por mucho tiempo. En realidad, se encuentra aletargada como si hubiera sido drogada para que no atendiera a nuestra llamada. Pero existe un antídoto que, más tarde o más temprano, acabaremos encontrando. Y será entonces cuando comience la auténtica aventura, nuestra entrada por la puerta falsa podría convertirnos en peligrosos espias indústriales; pero la realidad es otra, es el propio fabricante el que fomenta el deseo del usuario por descubrir la clave que esconde de la llave que nos permita entrar en la fortaleza protegida.




El sistema Turbo: un sistema roto.




El Decathlon «la meta».

## Los artífices de una desprotección

Pablo, Santiago y Emilio, decidieron aceptar el reto de ser los primeros en desproteger un conocido programa: «Decathlon», pero cometieron un error, quizá guiados por la alegría lógica de haber conseguido romper la «muralla Turbo», decidieron bautizar su éxito y poner su nombre en el programa. Lo que ellos no sabían es que, pasado algún tiempo, iban a aparecer gran cantidad de cintas en las que se podía leer su nombre, y lo que era más grave, éstas se estaban vendiendo en comercios dedicados a la venta de Software.

De esta forma, lo que empezó siendo un sano deseo de superar un reto, se había acabado convirtiendo en una actividad ilegal por culpa de aquellos que, sin escrúpulos, aprovecharon su amistad con ellos para conseguir copias que más tarde iban a llegar a los piratas comerciales.

La piratería sigue siendo un grave problema. La protección se están convirtiendo, poco a poco, en un distintivo de la calidad del programa. Estamos llegando a un punto en que antes de saber de qué va el programa, nos interesamos por el sistema de protección que lleva.

## Sistema Turbo

El sistema Turbo es el que se ha hecho más famoso de todos, hasta el punto de que, hoy día, se hace una distinción entre los programas que incorporan este sistema y los que no. Para entenderlo mejor, se ha convertido un poco en el coco del mercado de software. El sistema, en realidad, se basa en una doble protección: a nivel Basic y de código máquina.

Dentro del sistema de protección, junto a los conocidos métodos de invisibilidad de tinta y papel, la creación de líneas Ø y la incorporación del sistema de carga rápida, hay otros tan sofisticados, como hacer el programa inmanejable desde el BASIC, o crear una línea por encima de la 9999, la máxima que admite este sistema. La rutina de carga rápida tiene además, diversas protecciones en su ejecución, en distintos puntos de la misma. Otro detalle importante es que no existe Basic en la parte de código máquina, por lo que resulta inútil buscar ningún listado.

Sin embargo y, a pesar de todo esto, nuestros amigos de Zaragoza fueron dando con las claves, una por una, hasta lograr desenmascarar el misterio, y no sólo eso, además, según ellos, han logrado mejorar el sistema de protección Turbo.

Cuando se pusieron en contacto con nosotros, mantuvimos una larga conversación con uno de los componentes del grupo, el cual insistió en ocultar su identidad. Durante el transcurso de nuestra ENTREVISTA hizo hincapié, en repetidas ocasiones, en el hecho de que habían afrontado el reto de forma sana, sin ningún afan de lucro, y mucho menos de pirateo. Lo que si es cierto es que no hay ninguna ley que prohiba a un usuario coger el programa que ha comprado y hacer con él lo que le venga en gana, siempre y cuando no lo utilice con fines comerciales.

Hay que terminar insistiendo en la necesidad imperiosa de abaratar el software, como única medida que acabe, de una vez por todas, en el pirateo. Cuando deje de resultar rentable dejará de existir.

A continuación publicamos una carta de Pablo, Santiago y Emilio en la que se dan las pistas necesarias para que los más entendidos descifren el misterio, y los menos entendidos comiencen a interesarse por el tema.

## EL SISTEMA «TURBO» EN CLAVE

Intentamos describir telegráficamente, sin orden lógico, algunos de los puntos en que se apoya el sistema. Anticipamos que tiene un doble sistema de protección:

a) A nivel de basic.
b) A nivel de código máquina.

Los puntos en que se basa son los siguientes:

1.º) El BASIC cargador contiene enmascarada la rutina de carga rápida, junto con los datos de carga del programa.

2.º) La rutina de carga representa la última línea (aparente) del programa BASIC. Su número de línea es superior a 9999, por lo que no se ejecuta.

3.º) La rutina de carga está enmascarada con la función OR exclusive, de modo que cada octeto original se obtiene de hacer el OR exclusive entre dos octetos adyacentes del código enmascarado.

4.º) La rutina específica para desenmascarar la rutina de carga y trasladarla a su zona de ejecución, ocupa la zona de variables del BASIC. Su comienzo está en la dirección a la que apunta la variable VARS.

5.º) La rutina de desenmascaramiento está, a su vez, enmascarada mediante una utilización especial del registro indexado IY. Este registro se utiliza normalmente como registro doble de 16 bits. Los códigos son los mismos que los de los registros H y L, salvo con un FD delante de cada código.

6.º) Todas las líneas BASIC tienen número de línea Ø y PAPER Ø, INK Ø, ocultos o implícitos en cada línea. Ello los hace invisibles en el listado.

7.º) Las líneas BASIC no aportan nada esencial, salvo la 4.ª, que contiene oculta, en los códigos hexadecimales que siguen al indicativo ØE, la verdadera asignación en la pila de máquina directamente, de la dirección de retorno en caso de error, y que es la contenida en la variable VARS.

8.º) La puesta en marcha de todo el proceso, tiene lugar al finalizar la ejecución del BASIC, momento en el que en lugar de imprimir en pantalla el mensaje O.K., salta automáticamente al comienzo de la rutina de desenmascaramiento.

9.º) La rutina de carga rápida tiene protecciones diversas en su ejecución. En distintos puntos de la misma, comprueba los resultados de la carga y salta a la dirección Ø de la memoria ROM si no son los esperados. Antes de saltar al comienzo del juego, se borra ella misma.

El programa DECATHLON, además, tiene una protección adicional en su código máquina:

a) No tiene BASIC. Es inútil buscar ningún listado.
b) Sólo tiene definidas las variables del sistema que necesita el juego, como la de BORDER; por ejemplo.
c) Al no estar definida la zona BASIC (puede estar definida pero vacía), el programa es inmanejable desde el BASIC; sólo puede examinarse con un monitor de código máquina o similar (Acondicionar el BASIC supondría destruir parte del juego).

# INTERIORIDADES Y FUNCIONAMIENTO DE LA ULA (II)

*Primitivo de FRANCISCO*

**Adentrándonos en la filosofía de funcionamiento de la ULA, tratamos, en esta segunda parte, de cómo se direcciona eléctricamente, el significado de sus bits y de la generación de las señales de vídeo (Sincronismo y color).**

La ULA es para el microprocesador como un periférico más, y como tal, lo habilita con la dirección de puerto FEH o 254 endecimal, esto quiere decir que siempre que se accede a puerto con el bit AØ del bus de direcciones puesto a cero, la ULA se da inmediatamente por enterada y pasa a atender al microprocesador.

Eléctricamente, este direccionamiento se efectúa mediante un transistor que, ingeniosamente conectado, hace las veces de una puerta OR, tal como se muestra en el dibujo. Este transistor es el que en la versión «TWO» está colocado físicamente encima del Z-80. Esta puerta OR tan sui-géneris, tiene como entradas la señal IORQ del Z-80, que da paso a los periféricos, y el bit AØ que identifica a la ULA, como ya hemos mencionado. Analizando funcionalmente este montaje, se observa que únicamente cuando IORQ y AØ son cero, la salida por emisor del transistor también se hace cero. Esta salida es precisamente la que habilita a la ULA por su terminal denominado IRQGE.

Cuando se accede a la ULA, la mayoría de los bits del bus de datos tienen significado propio. (Excepto los bits 5 y 7 que no se usan).

Veamos a continuación sus respectivos cometidos:

— Los bits AØ, A1, A2, A3 y A4 son utilizados en modo lectura para recibir el contenido del teclado, como se vio en la semana pasada.

— Los bits AØ, A1 y A2 son utilizados en modo escritura para entregar, por las salidas de vídeo de la ULA, el color del borde que ha sido determinado por software.

— Los bits A3 y A4 en modo escritura y según su combinación binaria, determinan la función que se desea obtener, dicha combinación ha de ser también establecida por programa.

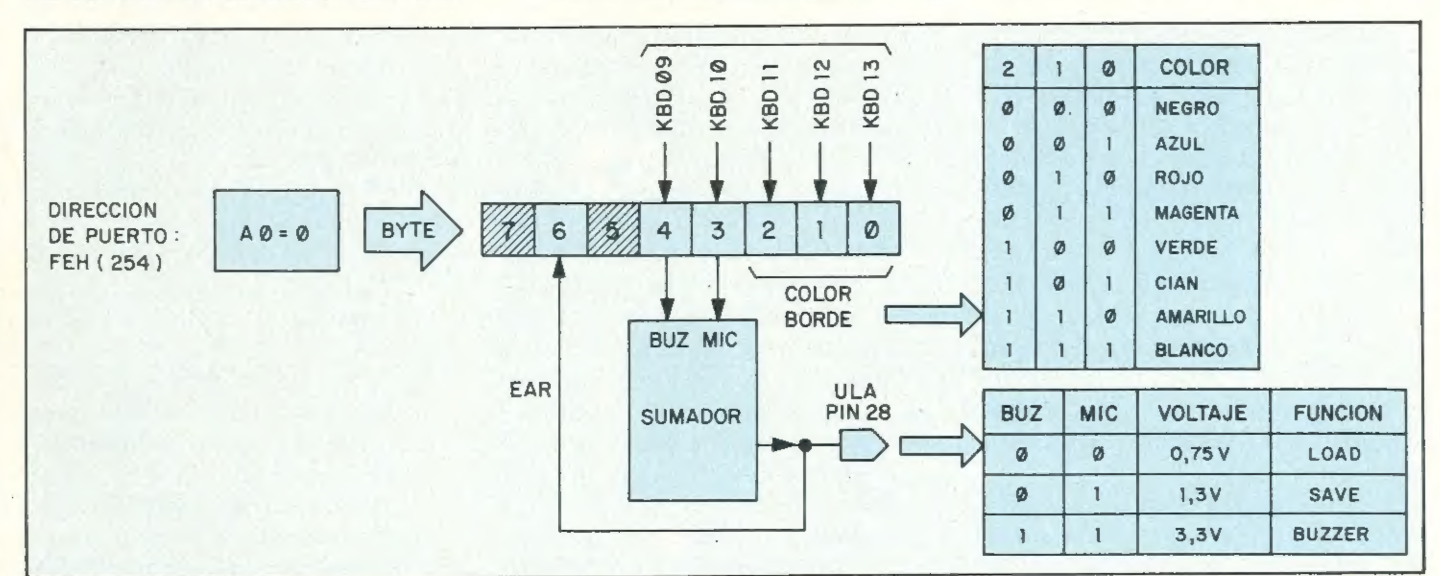
Veamos más detalladamente este concepto. De modo independiente, A3 actúa sobre la salida MIC para entregar la señal a grabar en cassette cuando se está en la función SAVE; dicho de otro modo, por este bit salen en serie, bit a bit, hacia la grabadora todos los programas que Vd. salva.

Por el bit A4 se actúa de modo independiente sobre el buzzer o minialtavoz interno. Este bit es utilizado para entregar sonidos y música al exterior; pero estos dos bits, aparentemente independientes, están ligados entre sí mediante las combinaciones que se muestran en la figura número uno. La razón de esta ligazón es la de obtener distintas tensiones eléctricas mediante una red sumadora en el pin 28 de la ULA, así, cuando A4 y A5 están a cero, la tensión únicamente alcanza 0,75 voltios; cuando son Ø y 1 respectivamente, la salida entrega 1,3 voltios, voltaje suficiente para activar la entrada de un cassette pero insuficiente para hacer sonar el buzzer.

Cuando A4 y A5 son uno, se alcanzan 3,3 voltios, con los que sí suena el buzzer.

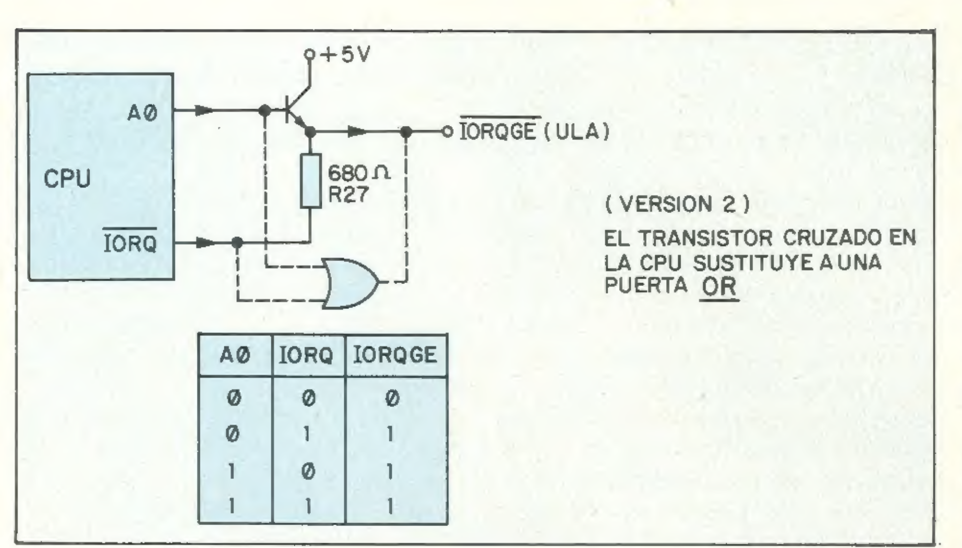
Esto ha sido así concebido para que únicamente suene el buzzer cuando se desea y no cuando se está en la operación SAVE, en cuyo caso los 1,3 voltios máximos son insuficientes para rebasar la tensión de umbral de los dos diodos que lleva en serie el buzzer (cada diodo tiene un umbral de 0,7 V. aproximadamente, por lo que hay que superar 1,4 V. para hacerle sonar).

— El bit 6A sólo tiene la misión de aceptar en modo lectura la señal que se recibe del cassette con el comando LOAD. Por este bit entran en memoria en serie bit a bit, los programas que Vd. carga.




Desglose de los bits del byte que direcciona a la ULA (puerto FEH) (Fig. 1).






Detalle de la funcionalidad lógica que configura el transistor que direcciona a la ULA conjuntando la señal IORQ y el bit AØ del bus de datos. (Fig. 2).

## El circuito de vídeo

En la figura número tres se muestra el circuito completo para el tratamiento de las señales de vídeo, desde que son generadas por la ULA, hasta su salida hacia la antena del televisor o hacia la entrada de vídeo de un monitor.

La ULA entrega por sus pines 15, 16 y 17 la totalidad de las señales de vídeo. (Denominadas por Sinclair U, V e Y respectivamente).

La señal U contiene la información de los colores R-Y (Red-Yellow), rojo amarillo. Esta es una de las dos señales compuestas de color para el sistema P.A.L.

La señal V contiene la información de los colores B-Y (Blue-Yellow), azul-amarillo. Esta es la segunda señal normalizada para la transmisión de color P.A.L.

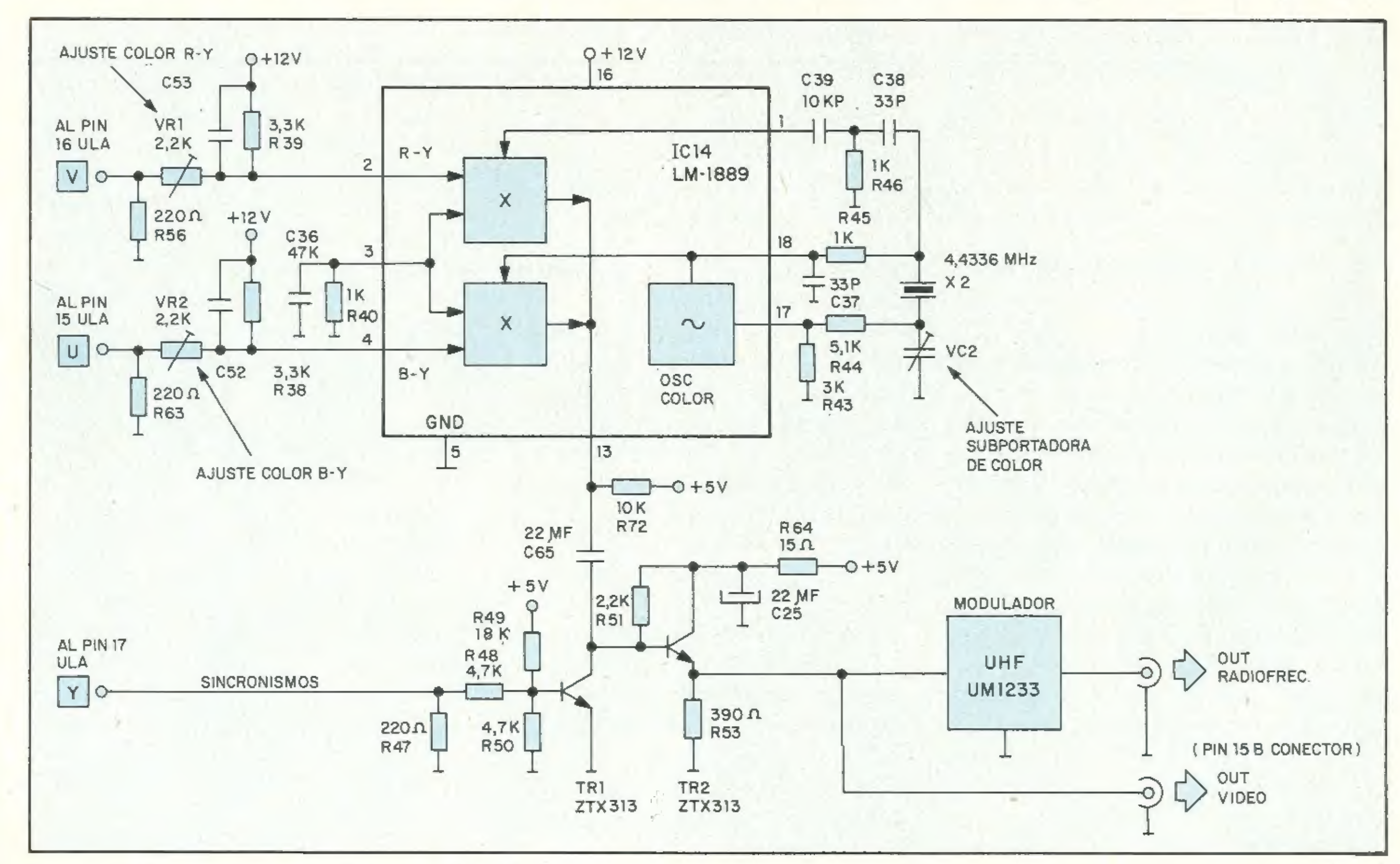
La señal Y contiene la luminancia o nivel de luminosidad para el color de cada pixel de pantalla. Recordemos que cada color ha de tener un brillo adecuado en función de la sensibilidad del ojo humano.

Si todos los colores fueran presentados con igual intensidad, veríamos más luminosos los verdes-azulados, pues nuestros ojos poseen mayor sensibilidad en estos tonos, quizá debido a que son éstos los que abundan con mayor profusión en la naturaleza.

La señal Y contiene también los impulsos de sincronismo. Para poder presentar una imagen en una pantalla, es preciso dirigir eficazmente al haz catódico que excita el fósforo de la pantalla. Hay que decirle cuándo ha de ir de izquierda a derecha mediante los impulsos de sincronismo de línea, y cuando de arriba abajo mediante los impulsos de sincronismo de cuadro.

La frecuencia de los primeros es de 15625 Hz para conformar las 625 líneas en que se barre la pantalla distribuidas en 312,5 por cada campo. El concepto de campo fue introducido para aminorar el efecto de parpadeo de la imagen. Cada campo barre líneas alternas en períodos consecutivos.

El tiempo que se tarda en barrer la totalidad de la pantalla es de 20 milisegun-

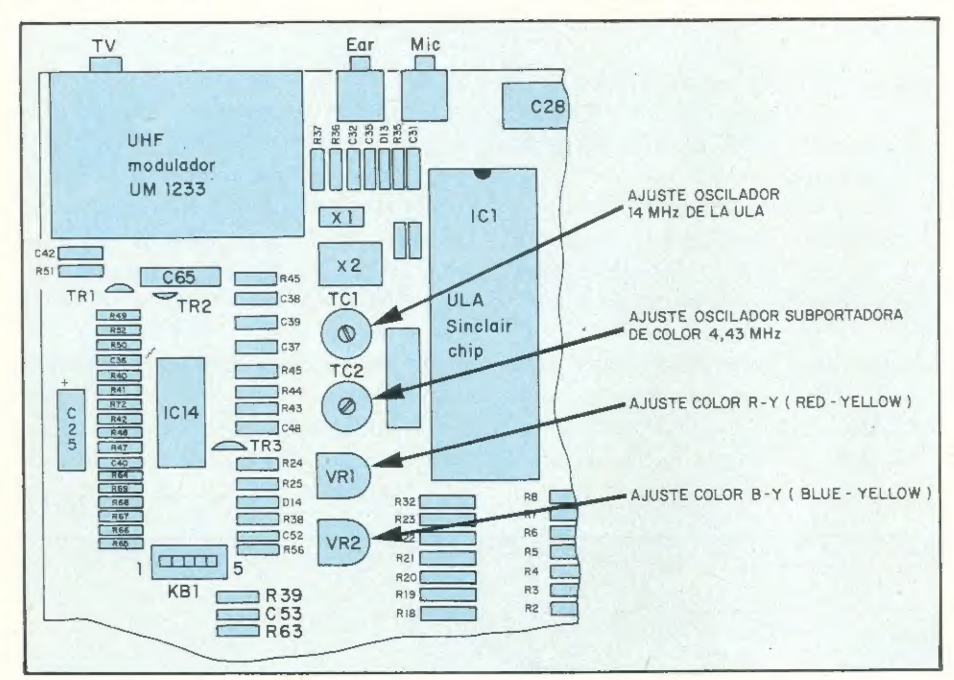



Esquema del área del vídeo del Spectrum. (Fig. 3).

dos, lo que da una frecuencia de 50 Hz, es precisamente con esta frecuencia con la que se presentan los impulsos de sincronismo de cuadro.

Las señales U, V e Y contienen, como hemos visto resumidamente, la información de color, luminancia y sincronismo. La ULA va controlándolas pixel a pixel sincronizadamente con el haz catódico que nos hará ver la imagen almacenada bit a bit, en el área de RAM del fichero de pantalla.

Las señales U y V, si bien aportan la información de color, no contienen la frecuencia de la subportadora de color necesaria para transportarlo mediante su modulación en fase.




**Detalle de la ubicación de los puntos de ajuste (versión TWO únicamente) (Fig. 4).**

## El circuito integrado LM-1889

El LM-1889 es un circuito integrado desarrollado por NATIONAL INSTRUMENTS para obtener la modulación en fase de la subportadora de color.

El LM-1889 contiene más funciones de las aprovechadas por Sinclair, además de generar la subportadora y modularla en fase, contiene también otro oscilador para proporcionar la segunda subportadora para el canal de sonido. También puede utilizarse mediante un tercer bloque como modulador de alta frecuencia para entrar directamente por la antena, pero estas dos últimas posibilidades están anuladas, el sonido eliminado vía atena y el modulador sustituido por uno incluido en una caja blindada de la cual sale al exterior la señal de antena.

El LM-1889 acepta las señales compuestas de color R-Y y B-Y, por lo que el acoplamiento con la ULA es el adecuado.

En la versión «TWO» existen unos potenciómetros para ajustar la pureza de color o, lo que es lo mismo, para ajustar la pureza en pantalla de los tres colores básicos, rojo, verde y azul. Esto se consigue mediante la actuación delicada de ambos potenciómetros.

En las versiones posteriores, estos ajustes manuales se han suprimido confiándose a la ya sobrecargada ULA que lo hace automáticamente, este método exigió una nueva versión de ULA. Los resultados, no obstante, son satisfactorios.

El objeto de eliminar puntos de ajuste va orientado hacia una mayor fiabilidad y estabilidad a la larga, una más barata elaboración al llevar menos proceso de fabricación y, sobre todo, eliminar la posibilidad de que manos inexpertas desequilibren el delicado cromático cuya puesta a punto requiere aparatos de medida al efecto.

El generador de la subportadora de color interno se encuentra controlado por un cristal de cuarzo de 4,4336 MHz que es la frecuencia de subportadora del P.A.L.

En la versión «TWO» existe un condensador de ajuste que, actuándolo manualmente, lleva al cristal hacia su frecuencia nominal exactamente. En las versiones posteriores no existe este ajuste, confiándose su exactitud a la baja tolerancia de los componentes que conforman el oscilar.

Por la pata número 13 del LM-1889 sale ya la subportadora de color modulada en fase, gracias a los dos bloques internos marcados en el esquema con una X (figura 3). Estos modulares de fase operan excitados por las señales R-Y y B-Y ya mencionadas.

El transistor TR1 de la figura 3, actúa de mezclador; ya que suma sincronismos y luminancia procedentes de Y (Pin 17 de la ULA) con la subportadora de color modulada en fase que entrega el LM-1889.

Así, en el colector de TR1 existe ya la señal compuesta y completa de vídeo.

El transistor TR2 actúa de adaptador de impedancias, su salida se bifurca hacia el modulador y hacia el terminal 15B del conector trasero. Para la conexión del Spectrum a un monitor de vídeo de color, existen dos posibilidades. Por una parte, mediante la señal compuesta del Pin 15B denominada «Vídeo», o mediante las tomas directas de la ULA (U, V e Y) que salen también por el conector trasero (U por 18B, V por 17B e Y por 16B).

Este segundo método es el utilizado por los monitores más caros y de mayor calidad.

## El modulador de vídeo

El Spectrum emplea un modulador de vídeo comercial para UHF código UM-1233.

La misión del modulador, es generar una frecuencia de UHF canal 32 aproximadamente, al cual corresponde una frecuencia de 560 MHz. Esta frecuencia es modulada en amplitud por la señal compuesta de video que existe en el emisor de TR2. La señal de alta frecuencia, una vez modulada, sale por el conector coaxial posterior del Spectrum para, vía antena, ser introducida en el TV doméstico.

El empleo de tan altas frecuencias en el modulador (560 Megahercios) obliga a tomar medidas especiales, como blindajes herméticos, pistas de circuito impreso anchas, ajustes de los pasos intermedios del modulador, etc.; el blindaje del modulador del Spectrum tiene la doble misión de no ser afectado por las radiaciones del resto de la tarjeta y no afectar él, a su vez, al funcionamiento de la placa.

Por último, recordar que la salida del modulador es de 75 ohmios, lo cual obliga a emplear cables apantallados de esta impedancia y a utilizar adaptadores e impedancia para TV de entrada de antena de 300 ohmios, estos adaptadores se encuentran fácilmente en el comercio.

# CONSULTORIO

## 80 caracteres

*En primer lugar felicitaros por vuestra revista, da una relación precio-calidad excelente (yo diría que la mejor del mercado), y sin entretenerme más, voy a plantearos mis preguntas que son varias y espero me contestéis a todas.*

*1. En el número 2 recomendáis a un usuario del Spectrum una impresora de 80 caracteres por línea. ¿Podría sacarle jugo a esos 80 caracteres si el mismo sólo sitúa 32 por línea? Si es así, ¿cómo podría hacerlo?*

*2. Qué ventaja tienen los monitores expeciales para micros si son más caros que las televisiones (y eso que se ahorran la circuitería de radiofrecuencia).*

*3. He visto que varios programas de los que se publican usan de una instrucción de la forma: 100 RANDOMIZE USR 6700, con otros números. Por más que he consultado el manual y un libro de código máquina, no he encontrado referencia a este tipo de instrucción. ¿Qué significa?*

*4. Por último, estoy pensando en suscribirme (no lo he decidido todavía porque estoy pendiente de la mili). Me podéis decir escuetamente, ¿qué programa contiene la cinta UTILITIES que ofreceis al que se suscribe?*

**Chini ECHEITA-Las Arenas (Vizcaya)**

□ 1. Aunque el ordenador sólo trabaje con 32 caracteres por línea en la pantalla del televisor, el número de caracteres por línea que aparecen en el papel de la impresora es algo que usted controla por programa.

2. Los monitores para ordenador tienen la ventaja de que la visualización es mucho más nítida y clara que en un televisor, lo cual se hace especialmente patente en el caso de aplicaciones que hagan uno extenso del color.

3. La instrucción a la que usted hace referencia es uno de los métodos que existen para conseguir ejecutar, desde el basic, una subrutina en lenguaje máquina; en el caso concreto que nos dice, el ordenador comenzaría a ejecutar el código existente a partir de la posición de memoria 6700 inclusive.

4. Puede leer la relación de programas de la cinta utilities en la página 16 del número 3 de nuestra revista.

## Varias dudas

*1. He instalado en mi Spectrum un interruptor normal y corriente, entre la fuente de alimentación que va del transformador a la entrada del ordenador. ¿Es peligroso para el aparato? ¿Qué diferencia hay entre esto y el reset? ¿Qué ventajas aporta el reset manual del que habláis en vuestra revista?*

*2. He observado en varias revistas el teclado profesional, en el cual el Spectrum va introducido dentro; pero en algunas tiendas existe otro similar, en el cual se desmontan los circuitos y se colocan en este otro teclado conectando las cintas del teclado de éste, por lo que el interior es más amplio y queda más ventilado ¿No es mejor este sistema?*

*3. Hace tiempo que busco por distintas tiendas juegos de inteligencia en los que no intervenga la habilidad, pero salvo el de Ajedrez no encuentro, sólo tuve el Hobit. ¿No lo hay en castellano? ¿Cuándo importarán dichos juegos? (traducidos, claro).*

*4. ¿Es preferible la unidad de disco de Spectrum, o se puede optar por otros modelos mejores?*

*5. He tenido oportunidad de observar juegos en otros ordenadores (Atari 500) y la diferencia existente entre éstos y los del Spectrum es comparable a la diferencia entre una película de dibujos animados y otra normal. ¿«A qué se debe esta diferencia? ¿Es debida a las limitaciones del Spectrum 48 K?*

**Luis MARTINEZ - Barcelona**

□ 1. Aunque para el usuario el efecto sea el mismo, desde el punto de vista lógico, el reset no equivale a cortar la alimentación, sino que supone forzar un salto del microprocesador a la rutina de inicialización. La instalación del interruptor no es peligrosa, siempre que las conexiones estén hechas con cuidado.

2. El sistema que Vd. apunta es, desde luego, mejor para la ventilación. Su inconveniente radica en que al desmontar la tarjeta de circuitos del Spectrum se corre siempre el peligro de dañarla.

3. Juegos de inteligencia para el Spectrum los hay a patadas. Desgraciadamente, el mercado español no parece demandarlos mucho y por ello los importadores no se molestan demasiado en importarlos, y mucho menos en traducirlos. De todas formas, la situación está cambiando y ya pueden encontrarse algunos programas de aventuras en castellano.

4. Si con unidad de discos del Spectrum se refiere al Microdrive, a nosotros nos parece un periférico que, por precio y prestaciones, se acopla perfectamente al Spectrum. No cabe duda de que las unidades de disco «reales» son mucho más potentes, pero hay que pararse a pensar si es realmente justificado comprar un periférico que pueda llegar a costar más del doble que el microordenador al que va a acoplarse. Como regla general, si de verdad cree Vd. que necesita discos, lo que ocurre es que necesita un ordenador más potente (y caro).

5. Uno de los grandes problemas del Spectrum es que sólo puede ofrecer dos colores a la vez en el mismo carácter, lo que limita sus posibilidades gráficas. La contrapartida es que, al reducirse la memoria de pantalla, queda mucha más memoria. De hecho, el Spectrum dispone de más memoria «real» para el usuario, algo más de 41 K, que otros ordenadores que se anuncian con 64 K de RAM, y en la práctica no llegan a las 40 K libres.

## Aclarar diferencias

*Por ser usuario de un Spectrum de 48 K he comprado vuestra revista, la cual me ha gustado mucho, teniendo además la ventaja de que es semanal y no mensual como lo son todas.*

*Me gustaría que me dierais solución a un par de problemas que se me han planteado:*

*1. ¿Qué diferencias existen entre un Interface Joystick y un Interface Joystick programable?*

*2. He intentado pasar el programa de la revista número 1, MICROCOPI, pero por mucho que lo intento me sale un error en la línea 9070 (2 variable not found, 9070: 1). Por mucho que lo repaso, no encuentro ningún error. ¿Es correcto el listado de vuestra revista?*

**Alfred XIFREU ALVAREZ - Barcelona**

□ 1. A diferencia de lo que ocurre con otros microordenadores, el Spectrum no lleva incorporado de origen una salida para JOYSTICK. Varios fabricantes de periféricos se aprovecharon de la situación y lanzaron al mercado los interface correspondientes, pero como era de esperar, cada uno utilizó un protocolo diferentes. Por ello, hay en la actualidad varios tipos de interfaces como el KEMPSON, AGF/PROTECK, FULLER y el del INTERFACE 2 de Sinclair, cuya tardía salida le impidió convertirse en el estándar. Ante esta caótica situación, surgieron los interfaces programables que pueden adaptarse a todo tipo de juegos, incluso si éstos no estaban pensados para utilizarse con Joystick. El inconveniente de estos modelos, aparte de su precio, es que antes de utilizarlos hay que programarlos para adaptarlos al juego correspondiente. Su gran ventaja es el poder utilizarlos con cualquier programa.

2. En cuanto a su segunda pregunta, aprovechamos para contestar a las docenas de cartas similares que han llegado a nuestra redacción. Es nuestra costumbre revisar cuidadosamente todos los listados que publicamos. Desde luego, nadie es perfecto, y por otra parte también puede suceder que en la imprenta se produzca algún error, pero en el caso concreto de MICROCOPI, les aseguramos que el listado es totalmente correcto. En la mayor parte de las cartas que nos han llegado con una copia del programa, el error estaba en las líneas de DATA, por lo que recomendamos a nuestros lectores que empiecen su revisión por esas líneas. Con un poco de paciencia lograrán ponerlo en marcha.

## QL

*Poseo un ordenador Spectrum ZX y estoy interesada en el nuevo modelo de Sinclair, «QL». Por ello, me gustaría que me informarais sobre sus posibilidades y si a este ordenador pueden serle acoplados los mismos accesorios del Spectrum.*

**Paloma ZOFIO - Castellón**

□ Las diferencias entre el QL y el Spectrum son fundamentalmente las siguientes:

1. Mayor cantidad de memoria RAM (128 K).
2. Un microprocesador mucho más potente y capaz, por tanto, de soportar un software bastante más sofisticado.
3. Dos microdrives incorporados de, aproximadamente, 100 K formateadas.
4. Un lenguaje Basic mucho más evolucionado y completamente estructurado.

Los periféricos del Spectrum son perfectamente compatibles con el QL.

## Para Víctor Contreras

*Felicitándoles por su estupenda revista, quiero ayudar a Víctor Contreras, que aparecía en el apartado «Consultorio» de MICROHOBBY, pidiendo un poke para anular la autorrepetitividad de las teclas.*

*Propongo un método con un solo poke:*

*POKE 23561,0*

*Con NEW se destruye este efecto.*

**Alberto ARCAS TRIGUEROS - Cáceres**

## Spectrum musical

*Pienso que uno de los inconvenientes que tiene el Spectrum es su sonido, por ello, quisiera saber si existe en el mercado algún periférico que lo convierta en un sintetizador de música y si hay programas musicales.*

**Joseba MARTINEZ DE MORENTIN**
**Gasteiz (Vitoria)**

□ Para suplir la deficencia de sonido que tiene el Spectrum, los distintos fabricantes de periféricos han creado diversos tipos de amplificadores que intentan, de algún modo, subsanar el problema. Sin embargo, no existe un dispositivo especial que convierta a su ordenador en un sintetizador, propiamente dicho. Sí lo hay, sin embargo, de voz.

Hay actualmente en nuestro país bastantes programas musicales, los más recientes son los que ha lanzado la casa Music Soft.

## Spectrum y T.V.

*Poseo un Spectrum 16K. Me gustaría saber si dicho ordenador puede dañar al televisor de alguna forma.*

*El cable que utilizo para conectar el ordenador a la televisión no es el que dan con el ordenador, ya que la T.V. tiene dos orificios para la antena del UHF.*

**L.M. - Zaragoza**

□ El ordenador no puede dañar al televisor de ninguna manera. El cable que usted nos propone es perfectamente adecuado.

## Conseguir más de dos colores

*¿Cómo podría poner más de 2 colores en una sola celdilla de atributos? Lo máximo que he conseguido son 2, utilizando el PAPER Y el INK. Sin embargo, tiene que haber alguna forma, pues he visto presentaciones de programas que lo tienen.*

**Flores TORRES - Cádiz**

□ Efectivamente, es posible realizar esto; le remitimos al manual del ordenador donde viene claramente explicado un programa Basic que solucionará su problema, concretamente en la página 112 del manual.

# DE OCASION

• VENDO Spectrum 16 K con instrucciones, cables, fuente de alimentación y más de 100 programas como: Jet Pac, Psst, Tranz AM, por 30.000 pts. Interesados llamar al 93/345 48 59. Barcelona. Ernesto Jordá Muñoz.

• VENDO Spectrum 48 K, interface 1, ZX microdrive, 2 cartuchos interface. Comcon para Joystick, y Joystick. Regalo programas y revistas. Todo prácticamente nuevo. Precio a convenir. Agustín Corral. Avd. Gran Bretaña, 2. Leganés. Tfno.: 686 40 85, (tardes y noches).

• VENDO ZX Spectrum 16 K, accesorios, manual castellano, garantía y regalo 100 programas 16/48 K. Buen estado, 29.000 ptas. Llamar a partir 6 tarde. Telf. 255 67 73. Madrid.

• VENDO órgano CASIO VL'I. En perfecto estado, con calculadora y memoria para grabar canciones por sólo 2.800. (Precio real 6.995). Interesados llamar 446 67 61, de 6,30 a 7, sólo Madrid. Excepto domingos. (Con este dinero, doy la entrada para el Spectrum 48 K).

• VENDO amplificadores para ZX Spectrum con accesorios, 1.500 pts. Alberto, Plaza Mosén Clapes, número 7. Primero. Primera. Telf.: 345 36 39.

• VENDO consola videojuegos Philips G7400+ con un videopac de juegos, impecable, con garantía. Precio: 28.000 ptas., ponerse en contacto con Félix. Telf. (986) 29 82 08, de 8 a 13 y de 15 a 18 h.

• CAMBIO consola de ATARI más 3 juegos y cable de alimentación. Máquina marcianitos «Galaxy II». Todo por un microordenador Spectrum 16 K. Interesados escribir a: Federico Rat Sánchez. C/ Lepanto 420, 2.º 4.ª 08025 Barcelona.

• VENDO ZX Spectrum 48 K nuevo, con todos sus accesorios y regalo programas comerciales (Hulk, Combat Linx, Sabre Wulf...) por 35.000 ptas. Nacho. Telf. (91) 255 47 97. También vendo cassette por 3.000 ptas.

• VENDO ZX81 con ciertas mejoras, por ejemplo: MICRO-INT ON OFF, Led indicador, fuente de alimentación estabilizada, etc. RAM de 32 K. Regalo 1 tomo de informática, varios programas, boletines de usuarios y 1 Stereo de automóvil. Todo por 20.000 pesetas. Dirigirse a: José A. Simón. P.º Lisboa, 13, 6.º 4. Fuenlabrada (Madrid).

• VENDO ordenador ZX81 en perfectas condiciones con adaptador de tensión, manual de instrucciones, 72 juegos y una cinta por 9.995 pts. Preguntar por Jesús al telf. (91) 404 99 76. Plaza Riofrío, 3, 9.º B. Madrid.